USER GUIDE

FORMSCAPE DEVELOPER™

Version 2.3
June 2000

F O R M S C A P E ®

e MPOWERING BUSINESS COMMUNICATIONS

First Edition, March 1999

Second Edition, June 2000

CONTENTS

OVERVIEW

This User Guide describes how to use the FormScape Developer module to create and save your FormScape projects.

The FormScape Developer module is used to design your new electronic form layout using graphical objects on a tree structure to process data. In conjunction with the Server module you can define the inputs and design and direct the output from it. During the design process you can also test your work by previewing it or printing it, so that you can see the results of your work as it progresses.

Installation of the FormScape Developer module is described elsewhere in the FormScape Installation Guide.

1 GETTING STARTED

1.1 INTRODUCTION

This section of the User Guide describes how to get started using the FormScape Developer module by creating a simple form design as an initial exercise. This is followed by a more detailed explanation of the FormScape project tree and how to move around it, followed by tutorial exercises using prepared sample data.

It is recommended that you select the 'Show HelpLinks' option from the Documentation Options dialog (select Documentation from the Options menu). These links allow you to display information on the object. The screen shots in this document have been taken with this setting enabled.

1.2 INITIAL EXERCISE

To gain experience and confidence in using the FormScape Developer module, the initial exercise described below is recommended. In it you will design a simple form layout comprising text and graphics using some of the forms editor features and print it as a sample form. The steps involved in carrying out this initial exercise are described in some detail, but the subsequent tutorial exercises are less so, making use of the experience gained in the initial exercise. You are of course free to experiment at any time as you work through the exercises by trying options and features of particular interest.

Carry out the following steps: -

1.   To start Developer, click on Start|Programs|FormScape2|Developer. The Developer logo appears, followed by a full-screen view of the project 'tree'.

2.   Click on the cross to the left of the First Action icon.

3.   Click on the cross next to the Pages icon.

4.   Click on the Form (Graphic) icon. A blank page with a grid at one-inch intervals appears in a Graphic editor window - this is the blank sheet on which you are to create your form design. A tool bar at the top of the screen shows available tools for editing/creating the form layout.

5. Click on the Zoom Out button in the tool bar until the entire blank page is displayed.

6. Click on the line draw button and draw a vertical line on the blank page by clicking on a start point and then dragging. On the tool bar above the Graphic editor window, click on the line thickness and line style boxes to select a line thickness of 2 pt and a dashed style.

7. Click on the box draw button in the tool bar and draw a rectangle on the page by clicking on a grid intersection and dragging down to the right to a further grid intersection, making a rectangle of three grid squares wide by two grid squares deep. Select a line thickness and style e.g. 1 pt and continuous.

8. Click on the text button and draw a rectangle on the page by clicking on a start point and dragging down to the right. Make this rectangle two grid squares wide by one grid square deep. A further window opens with a flashing cursor. Type in the word STATEMENT, and press Enter to finish, as instructed. The word STATEMENT now appears in the highlighted text box. Select a font and point size e.g. Arial 16 pt, by clicking on the appropriate tool bar menus above the Graphic editor window.

9. Click on the picture button and draw a rectangle on the top left of the page by clicking on a start and end point as before. Make this rectangle one grid square in each direction. A window opens for you to select a graphics file to insert in the highlighted rectangle. Select the file (Windows .bmp format) by browsing for *C:\program files\formscape2\examples\fslogo.bmp*. Double-click on the file name to insert the graphic in the highlighted rectangle. The FormScape logo should now appear on the page.

10. Click again on the box button and draw a rectangle on the page that is two grid squares wide by half a square deep. Select the line style and thickness to be continuous and 4 pt. Click on the box shadow button above the Graphic editor window. The Shadow window appears. Click on Draw Shadow and enter an offset of 0.1 inches. Click on the OK button. The rectangle you have drawn now has a drop shadow added to it. You can also select a color (the default color is black) for the shadow by clicking on the Set Color button in the Shadow window to produce a color palette.

11. As a further refinement, with this rectangle still highlighted, click on the rounded corners button above the Graphic editor window. The Rounded Corners window now appears. Click on "Draw rounded corners" and enter a size of 0.2 inches. Click on OK. Your drop shadow rectangle now has rounded corners added to it.

12. As a further extension of this feature, draw a rectangle one grid square, and select the rounded corners button as before. Select a size of 0.5 inches i.e. half the size of the rectangle in each direction. Click on OK. This square has now been transformed into circle. (Note that it also has a shadow added to it that can be removed if desired by unclicking the Draw shadow box in the Shadow window). A circle can alternatively be produced by clicking on the Ellipse button (next to the box draw button on the toolbar) and dragging a square on the form – the ellipse contained within the square now becomes a circle, since its vertical and horizontal axes are identical.

13. Click on the barcode box and draw a rectangle on the page that is one grid square wide by half a square deep. Right-click on this rectangle and select Explore from the floating menu. The screen now shows the project tree in the Exploring pane alongside the Graphic editor window.

    In the Exploring pane, notice that you have now added a branch called Barcode at the end of the tree. Click on the cross to the left of the Barcode icon on the tree: a sub-branch labeled 'Input =' appears below the Barcode icon. Click on the Input icon (pair of open and close quote marks) to open a Text editing box. Type in the numbers "12345" in this box, and click on OK. A barcode representing these numbers now appears in the highlighted rectangle on the Graphic editor window. Double-click on this rectangle to open the Bar Code Properties dialog box. Select Code 39 in the Symbology box, Horizontal in the Orientation box and Raw from the Human Readable drop-down list.

Click on OK. Now stretch the bottom of the highlighted barcode rectangle to one square deep to reveal below the barcode the numbers 12345.

14. In the tool bar, click on the printer button to perform a test print of the page. The form you have designed should look similar to that shown below, and incorporate the following features :-

    (1)   the text 'STATEMENT' in Arial 16 pt font
    (2)   a graphic (the FormScape logo)
    (3)   a blank rectangle with a thick continuous frame, drop shadow effect and rounded corners
    (4)   a blank rectangle with a continuous frame around it
    (5)   a vertical dashed line
    (6)   a barcode representing the numbers 12345
    (7)   a blank circle with a thick continuous edge

15. Click on the double-headed arrow on the top left of the Graphic editing window and select 'Remove Pane' to return to the Exploring project tree view. In the File menu at the top of the screen click on Save As and then the To the Server sub-option. A display box shows the root folder on the Server. Click on the root folder and open the sub-folder Processes. Enter a filename e.g. "Exercise1" and save your project by clicking on OK. The file is automatically saved to the server. You have now designed the sample form layout shown below using the Developer module of FormScape.
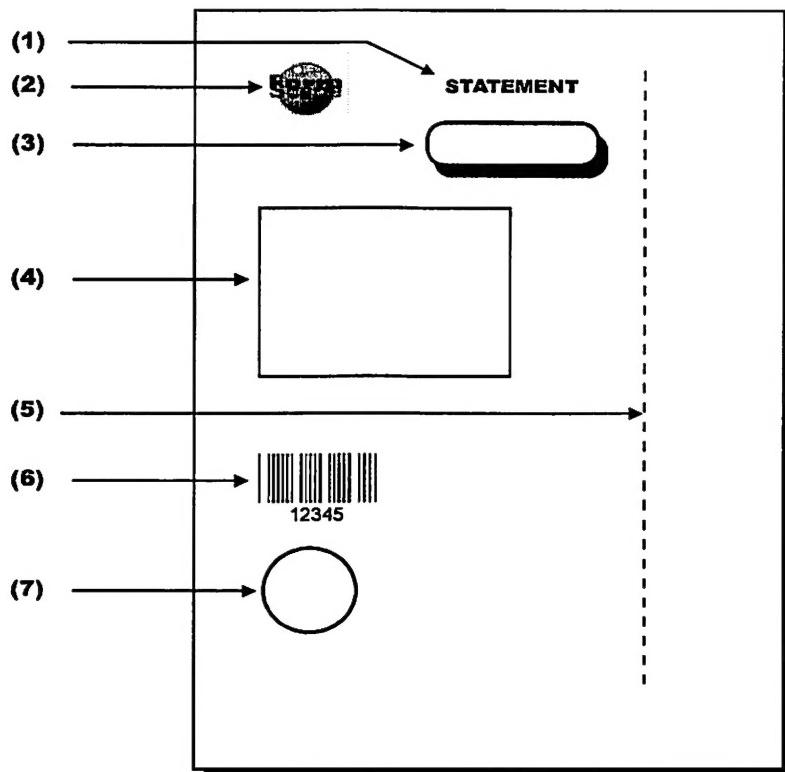
Sample form layout (exercise 1) showing some of the form editing features

1.3 "NAVIGATING" AROUND THE PROJECT TREE

As you have already discovered in the initial exercise, when you first run FormScape Developer, the project opens with a tree view in the Exploring pane. This view allows you to manipulate the project tree directly by adding further icons to add-points on the tree.

The top of the tree shows an icon (labeled 'Process') depicting the Queue Process for this part of the project - everything that you set up within the tree structure will be a function performed within this branch of the project.

You can start a new branch or open an existing branch using the File menu or the icons on the tool bar. You can save a branch you are working on using the File menu or the Save icon on the toolbar. Branches are saved with a .FSP extension.

1.3.1 WINDOW PANES

The view in the Developer module shows the menus and a tool bar at the top of the screen, and the tree as a full-screen view. During the development process you may want to see two or more parts of the tree at the same time, or you may want to simultaneously display your captured data and the form you are currently designing. You can divide the main screen into any number of sub-windows referred to as "panes".

The active pane (always distinguished from any others on the screen by a colored bar at the top) can be divided horizontally or vertically. The icon that allows you to do this is the small double arrow at the top left of the active pane. When you click this, a small menu drops down, allowing you to split the screen, remove the pane altogether, or change the view to small icons so that you can get more of the tree on the screen.
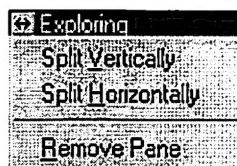
Alternatively, you can click the right mouse button when the cursor is in the active pane. A menu appears that allows you to split or unsplit the screen. In this case a vertical pane will split into two horizontal ones, and a horizontal pane will split to produce two vertical panes.

At the top right of each pane there is a small cross icon which allows you to restore the tree view in the active pane. This icon only appears if the pane is currently displaying anything other than the tree view.

The size of the individual panes can be changed by placing the cursor on the vertical or horizontal bar that splits the window, and dragging it to a new position using the left mouse button.

A quick method of re-arranging the panes on the screen is by dragging the bar at the top of the active pane into one of the others. This will change the layout of the existing panes on the window.

1.3.2   THE FORMSCAPE TREE

The key to using FormScape is managing the project tree by building up a number of connected objects (icons) which interact to perform the required functions. Many objects can be added to the tree using the high level editors, but you will also need to learn how to manipulate the tree directly to gain maximum benefit from FormScape.

1.3.3   WHY USE A TREE ?

FormScape makes use of a tree structure for holding all the objects in your project together. As FormScape is primarily a programming environment, any object placed on the tree receives its settings from other objects connected to it. By using the extensive linking ability of these objects, it is possible to alter the inputs to an object dynamically. For example, the color of a printed number could be linked to a value that appears within the data, for example only negative numbers could be printed in red, with all others black.

1.3.4   LABELLING THE TREE

Next to each object on the tree is a small text label. These labels are automatically assigned a default text as you add them to your FormScape project. It is strongly recommended that you edit these labels to suit your own requirements as you produce them. To alter a label, double-click on the text. A Rename dialog box opens, which allows you to change the text.

In addition, there is a 'Documentation' facility available for adding an optional comment to your object label, and this comment may be displayed as an additional green text alongside the label. To use this feature, select the label of the object to which you wish to add a comment by clicking on it, select the Documentation Viewer option from the View menu and enter the comment in the Documentation Viewer window. When you close this window your comment appears alongside the object's label. If you select the Documentation option in the Options menu, you can further choose whether to display the comment on the tree or not, and how many characters of the comment to display (in the range 5-200 characters). Note that a displayed comment will also be printed if you use the Print Tree option to print a tree view.

Note: The example diagrams in this document show the Documentation Option "Show object type hyperlinks" enabled (select Documentation from the Option menu).

1.3.5   EXPANDING THE TREE
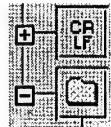
The tree can be expanded or contracted at any time so that you can arrange the various objects you want to work with on the screen. The tree structure has small square 'joints' which display either a + or a –. Clicking on a plus (+) joint will expand the tree, while clicking on a minus (-) joint will contract it.

1.3.6 OBJECT EDITORS

Each object on the normal tree view appears as an icon. There are two types of icon: depressed icons and engraved or flat icons. Clicking on an icon will open the editor associated with that icon, unless it is already open, then the editor window becomes active. The editors allow you to perform functions associated with that icon, or to alter the way it behaves. For example, clicking on a form icon opens the Graphic form editor in the current pane.

1.3.7 CHANGING THE TREE ORDER

It may be desirable or necessary to alter the order in which objects appear on the tree. Objects cannot be moved to a higher or lower level of the tree, but objects on the same level can be re-arranged into a different order.
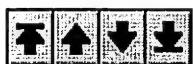
There are four icons at the top of the screen that allow the order of the objects to be changed: move to top, move up, move down and move to bottom. To move an object, click on its text label once with the left mouse button, and then click the appropriate tool to move it.

1.3.8 ADDING OBJECTS TO THE TREE

Objects are added to the tree in two ways. With a high level editor such as the Form Editor or the Page Mapper, objects are automatically added to the tree (as you design a form every piece of text that you add to the form is added as a text object on the tree). You can also add objects to the tree manually wherever you see a branch that has a small square stub ⊡ (an add-point). Clicking on the add-point displays a Create Object dialog, which contains an alphabetical list of available objects grouped into an alphabetical list of categories. Choose the object you want to add to the tree from the window. When the object is added it may contain a sub-branch of other objects (child objects) which are automatically added.

1.3.9 DELETING OBJECTS FROM THE TREE

It is also possible to delete objects from the tree. With a high level editor such as the Page Mapper or the Graphic form editor, select the object – for example, a rectangle – by clicking on it with the left mouse button, and then pressing the Delete key. However, if you want to delete objects from the tree directly, select the object on the tree by clicking the left mouse button on its text label and then press the delete key.

Note that when deleting objects you may receive a warning message saying that the parent object (the object above it on the tree) requires the object, or that the object is linked to another object and cannot be deleted. This is because the object you are trying to delete is essential to the correct operation of its parent object.

If an object is linked to another object it will be shown with a small yellow arrow pointing away from it. You will only be able to delete this object when you have removed the object it links to, which will be connected elsewhere on the tree.

1.3.10 CHANGING OBJECTS ON THE TREE

When you add objects to the tree, certain object structures or tree branches are added as defaults. The great versatility and flexibility of FormScape derives from the tree, which can be built and altered to perform functions that fulfill your specific requirements.

This flexibility is achieved because the various objects you build the tree with are interchangeable. For example, you may design a form which has a piece of text on it stating 'Cash Only', and you may want to add this text to some forms only but not others. You will need to change the *text* object on the tree into a *testing* object, which determines whether or not to print the text. In turn the testing object will need to be presented with some data from the current page in order to make its decision. In this way the project can grow and change to fulfill your precise requirement.

As previously mentioned, to change an existing object from one type to another, point to the object and click the **right** mouse button. Select 'Change Type' from the drop-down menu. You are presented with a list of compatible object types in a Change Object window, similar to the Create Object window.

Note: *Objects can only be changed to a 'similar' or 'compatible' type. For example, it is not valid to change a Printer object into a Scheduler*
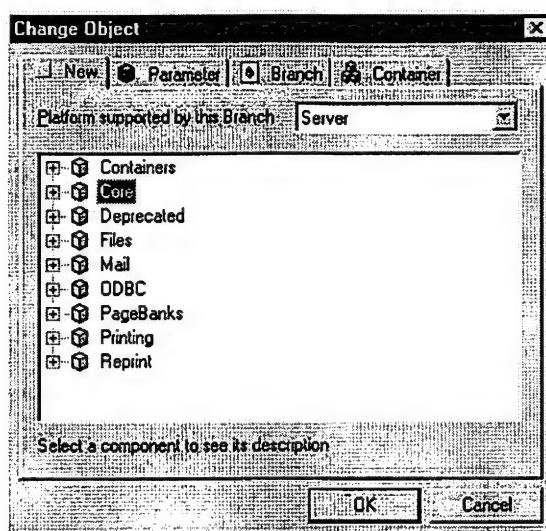
1.3.11   COPYING OBJECTS ON THE TREE

It is possible to copy objects from one part of the tree to another, but only if the target position is capable of accepting the object. For example, when you produce a 'Print Graphic' object on the tree, and you want to make a copy of it in order to produce a second sheet for another printer. (A Print Graphic object produces a sheet, normally with a form on it, and sends it to a specified printer.) To copy the object, place the cursor on the object you want to copy and using the **right** mouse button, drag it to the target part of the tree. You can copy the object to a different part of the tree in the same pane, or onto a part of the tree which is being displayed in another pane.
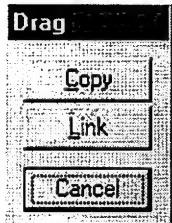
When you release the **right** mouse button, a small menu (the 'Drag' menu) appears allowing you to cancel the operation, or to 'Copy' or 'Link' the object. Selecting 'Copy' from the menu causes the selected object, and all the object's children (those objects attached below it on the tree), to be duplicated to make an identical copy. The new object is only identical at the time it is copied. It can then be altered independently of the original.

A linked object is not a new object. It is a reference to the original, and any changes made to the original are applied to all its linked objects.

Note that when dragging an object to a target position that is not an add-point, the object will always replace the target object.

1.3.12   COPYING OBJECTS BETWEEN PROJECTS

You may want to copy an object you have created into another project using a 'branch' file. **Right** click on the object you want to copy and select 'Save As' from the small menu. You then have the option of saving the object either to 'localhost' (the server) or your computer's hard disk ('My Computer'). Enter a file name, and the object you have selected (including all the child objects connected to it) is then saved as a FormScape branch file with the file extension .FSP. You can then paste this file into different projects by **right** clicking on an add-point in the project tree, selecting 'Insert Copy of', then either 'localhost' or 'My Computer' (as appropriate) and selecting the file from the list appearing in the Open Branch window.

It is important to note that some of the objects within the branch file may have been linked to other objects in the original project, which do not exist in the new project. These links are automatically removed when the branch file is made. Links to other objects within a branch file always remain intact.

2 MANAGING A PROJECT

The FormScape Server resides on a designated Windows NT4 Server machine, and the FormScape Server Service intercepts the print queues (file directories or mail accounts) looking for data that requires application intervention. It is completely transparent to the NT server and does not interfere with any print queues that have nothing to do with FormScape. The FormScape Server does not need to be installed or running on the same machine as FormScape Developer, it is recommended that it is installed on a machine or machines elsewhere on the network.

The diagram below shows how two Developers can work on projects that are saved and running on one Server. Note that there is a system to ensure that two Developers are not accessing the same branch simultaneously.
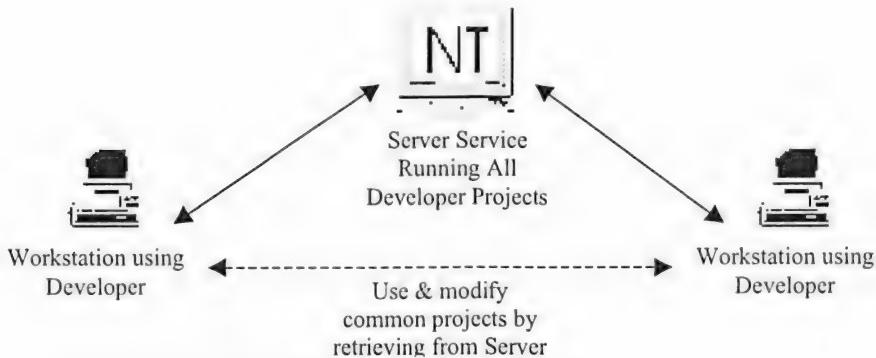
2.1 THE DEVELOPMENT ENVIRONMENT

FormScape Developer includes a fully functional FormScape Server (with spoiler) which allows you to test projects on your local machine. Your local machine can be referred to as either "localhost" or the name of your machine (e.g. X00187).

2.2 SAVING A PROJECT

FormScape projects can be held either on your computer's hard disk (To My Computer) or on the FormScape Server (To the Server at…) and all objects that are saved are referred to as a branch: a project can contain any number of branches.

Select Save As from the File menu to save your branch. Two options appear in the sub-menu; "To My Computer" or "To the Server at…". A project must be saved "To the Server at…" if you need the process to run. However, if you choose " To My Computer" your branch is exported to a location of your choice. This

method should only be used for development purposes, because the process only runs if saved on the Server. Note that a branch saved in this way will have the filename extension .fsp.

Typically, when developing a project, it would be saved locally to the Server for testing purposes. To do this select Save As "To The Server at...". The Server defined here should either be localhost or the name of your computer (e.g. X00187). If the Server name is not correct then you will need to change the Server. See "Changing the Server" for more information.

2.3   THE SERVER STRUCTURE

Once Save As "To The Server at..." has been selected the display will show the root folder on that Server (which has the name *name of computer. name of domain)*. To open a folder, double-click on the folder. Below that folder there will always be two folders which have special purposes, the Actions folder and the Processes folder. Only use the Processes and Actions folders if their functionality is required.
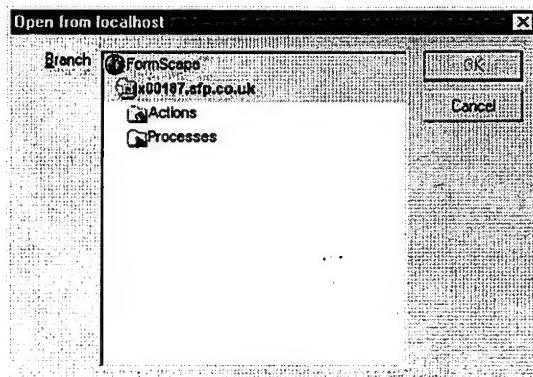
2.3.1 PROCESSES FOLDER

Any branch process that is saved to the Processes folder is automatically started. Note that by default the Input queue type is a Print Queue so for every Queue Process that is saved in this location a Printer will be created on your computer.

2.3.2 ACTIONS FOLDER

Any branch saved into the Actions folder is made available to the Viewer.

2.3.3 CREATING NEW FOLDERS

To create a new folder, double click on the folder where you wish to create your new folder. Right click on the folder and select New Folder. Enter a name when prompted.

2.3.4 MANAGING ITEMS ON THE SERVER

Any of the following functions are available from either the "Open from..." or "Save to..." dialog. Note that performing any of these functions may adversely affect any links between branches.

2.3.4.1   CUT

Right-click on a branch or folder and select Cut. Right click on the destination folder and select Paste. The branch will be deleted from the original location.

2.3.4.2   COPY

Right-click on a branch or folder and select Copy. Right click on the destination folder and select Paste. The branch will also remain in the original location.

2.3.4.3   DELETE

Right-click on the branch or folder that you wish to delete and select Delete.

2.3.4.4   RENAME

Right-click on the branch or folder that you wish to rename and select Rename.

2.3.5 BRANCH PROCESSES

A branch process is a branch that contains a process, and will be started automatically if the branch is saved in the Processes folder. Otherwise select Processes from the Server menu and log onto the server. Click Start and select the process that you want to start. Note that branch processes can be independently stopped and started – to stop a branch process highlight it and click on Stop. The Events window shows any current activity or error messages.

2.4 CHANGING THE SERVER

To be able to view another developer's project or to save a project to a universally available machine, we need to define which Server to log in to. To do this, select Change Server option from the Server menu. Enter the Server machine's name and it's Domain, and also your User name and password on that machine.

2.5 MANIPULATING PROJECTS OVER THE NETWORK

First change the Server as described in the previous section. To Open a project, from the File menu, select Open "From the Server at...". Locate the branch from the Processes (or other folder) on the named server machine. Note that if another Developer has a branch open, the branch icon will be displayed in red and it will not be possible to open the branch.

When a project needs to be saved to a universally available machine, it would be saved to the specified Server. To do this, from the File menu select Save As..

By selecting "To the Server at...", the project will be saved to that server machine and available to all Developers able to log in into that Server. Once selected the display will show the root folder on that Server (which has the name *name of computer. name of domain*). To open a folder double left click on the folder. To automatically start a branch process, save it into the Processes folder.

Note that when opening projects from a Server that is not on your local machine, different printer settings may have been used, or the printers may be unavailable to use. All print and page settings should be checked before running/saving the project.

2.6 BACKING UP YOUR SERVER

A project can contain any number of individual branches. It is possible to make a copy of all branches by saving each branch "To My Computer", but this is time consuming and difficult to manage. Using the backup function will save a folder at the root level and will save all sub-folders and directories below.

Access the Server by either selecting Save As "To the Server at..." or Open "From the Server at...". Right click on the Folder (or branch) that you wish to backup.
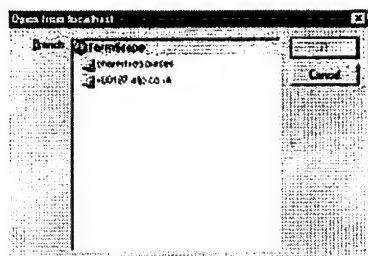
In this example, right clicking on the FormScape globe and selecting "Create Backup" will backup everything. Right clicking on x00187.afp.co.uk will backup everything held within that folder, but will not backup the "shared resources" folder.

To restore from backup right-click on the FormScape globe and select "Restore from Backup". Note that regardless of where you right-click to restore the backup, the folders and branches will be restored to their original location.

Note that the Processes and Actions folder only have their special functionality when they are within the home folder (the name for the home folder is taken from the computer name and TCP/IP domain name). Therefore, if you restore a backup from one machine onto another the contents of the Processes folder will not automatically start and the contents of the Actions folder will not be available from the Viewer. This issue is addressed in Section 4.12.

2.7   SECOND EXERCISE

This exercise is designed to demonstrate how to save branches and start processes on your local machine. As part of this exercise FormScape will create a printer called Invoice. If a printer already exists with this name, then it will be deleted.

1. To start Developer, click on Start|Programs|FormScape2|Developer. A project will be created with default objects. We will use this project without any enhancements for this exercise.

2. From the File Menu select Save As.

3. Select *To the Server at localhost* (the name of your local machine is acceptable instead of localhost. Otherwise Select Change Server from the Server menu and change to localhost – see section 2.4)

4. Open the Processes folder by double clicking on the Home folder and double clicking on the Processes folder.

5. Enter the branch name 'Invoice'.

6. Click Save. It may be necessary to log onto the FormScape server at this point. Note that when you are returned to the Developer the Queue Process name has now been changed to Invoice.

7. Check that the Queue Process has been started (select Processes from the Server menu).

8. Check that the Print Queue "Invoice" has been created (from the Windows Start menu select Settings/Printers).

The exercise above saved a process into the Processes folder so it would be automatically started. It will not always be desirable to automatically start all processes: in a development environment it may be necessary to create multiple branch processes and if the input type is a Print Queue (as is the default) then a printer will be created for each branch process that is saved. The next part of this exercise will describe saving the branch process into a different location and manually starting the process.

1. To start Developer, click on Start|Programs|FormScape2|Developer. A project will be created with default objects. We will use this project without any enhancements for this exercise.

2. From the File Menu select Save As.

3. Select *To the Server at localhost* (the name of your local machine is acceptable instead of localhost. Otherwise Select Change Server from the Server menu and change to localhost – see section 2.4)

4. Double clicking on the home folder to open it. Right click on the home folder and select "Create Folder".

5. Enter the Folder name Test.

6. Double click on the test folder that you have just created.

7. Enter the branch name 'Statement'. Click Save.

8. Check that the Print Queue "Statement" has **not** been created (from the Windows Start menu select Settings/Printers).

9. Check that the Queue Process has **not** been started (from the Server menu select Processes).

10. Click the Start button from the Processes dialog (select Processes from the Server menu). Locate and select the Statement branch and click OK. The process will now be started.

11. Check that the Print Queue "Statement" has been created (from the Windows Start menu select Settings/Printers).

Finally, you should delete both of these branches from the Server.

1. To start Developer, click on Start|Programs|FormScape2|Developer. A project will be created with default objects. We will use this project without any enhancements for this exercise.

2. From the File Menu select Open.

3. Select *From the Server at localhost* (The name of your local machine is acceptable instead of localhost. Otherwise Select Change Server from the Server menu and change to localhost – see section 2.4).

4. Double click on the home folder and Processes folder to display the Invoice branch. Right click on this branch and select Delete.

5. Double click on the Home folder and Test folder to display the Statement branch. Right click on this branch and select Delete to delete the Statement branch. To delete the Test folder, double click on the home folder, reselect the Test folder, right click on it and select Delete.

3   TUTORIALS

3.1   TUTORIAL PROJECT 1

In this first project you will develop a form layout using the Developer module. All the information needed to carry out this project will be saved on the server in a single FormScape file called Tutor1, which can then be modified and updated by the Developer module for the subsequent tutorial projects 2, 3 and 4.

In this first project you will design a simple form in the same way as in Exercise 1, but simulate adding host data to it by using a prepared demonstration text file (*dem.txt*).

Carry out the following steps: -

1. To start Developer, click on Start|Programs|FormScape2|Developer. The Developer logo appears, followed by a full-screen view of the project tree in the Exploring pane.

    When you start Developer a new project is automatically started with the default name 'Process', so it is useful at the outset to save your project under the name 'Tutor1'. Do this by clicking on the File menu on the top of the screen, selecting Save As and then the To the Server sub-option. A display then shows the root folder on the Server. Click on the root folder (which has the default name *'name of computer.name of domain'*) and open the sub-folder Processes. Enter the filename 'Tutor1' and click on Save.

2. Expand the tree by clicking on the small squares with a + sign ⊞ so that you can see the other objects connected to the tree. Note that the Process icon at the top of the tree has now been named Tutor1.

    The Print Queue icon is responsible for collecting the data from your host application. When the Server module is running it will create a shared printer in your Windows system called Tutor1.

    The next icon is the Language Interpreter, which takes data from the Input Queue, and breaks it down into pages, which are then stored in a FormScape PageBank.

    The next icon below is a folder labeled 'On Job', whose sole function is to hold together in one place all the actions you want to perform in developing this part of your project. It currently contains only a Print Graphic icon labeled 'First Action' (and its child objects).

3. Double-click on the text label next to the Print Graphic icon ('First Action') and in the Rename dialog, change the name to 'Top Copy'.
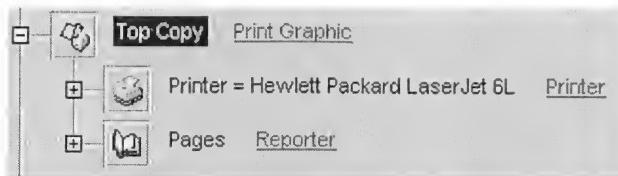
4.   If you have not already done so, expand the tree by clicking on the + sign next to the Print Graphic icon. Two further icons are now displayed – the Printer icon and the Reporter icon (currently labeled 'Pages') – that together are responsible for producing the printed output you require.

5.   Click on the Printer icon and select your target printer. The printer you choose will be used for the form you are currently developing, but you could assign a different printer to each form in your project.
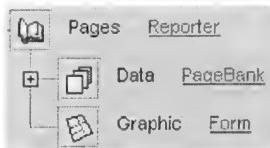
6.   Click on the + sign next to the Reporter icon to expand the tree further. Two more icons are displayed – a PageBank icon (currently labeled 'Data') and a Form icon (currently labeled 'Graphic').
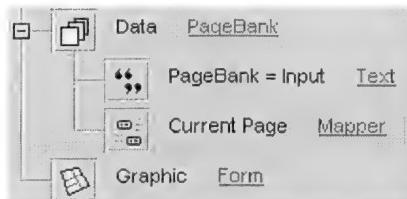
7.   Click on the small square with a + sign next to the PageBank icon to expand the tree further. The last two icons are now revealed as a Text object labeled 'PageBank = Input' and a Page Mapper (currently labeled 'Current Page'). These two icons control the PageBank: each page in the named PageBank is passed to a Page Mapper, which is used later on to place the host data on your form.

8.   To start designing your form layout, click on the Form icon to open the Graphic form editor. This editor uses a toolbar containing many of the same tools used in standard Windows drawing packages e.g. you can draw boxes, add text which can be formatted, change fonts, line widths and styles etc. When first displayed the blank page on which you are going to design your form is overlaid with a page border and a grid spaced at inch squares, which can be used as a guide to positioning the various objects you place on the form. For this project you will construct a simple form with two rectangular boxes, some text and a graphic (the FormScape logo).

9.   Click on the box tool on the tool bar and place the cursor on the intersection of the first
      horizontal gridline from the top of the page and the first vertical gridline from the left
      edge of the page. Press and hold the left mouse button and drag a rectangle that is 3 grid
      squares wide by 2 grid squares deep. Release the mouse button - a rectangle appears on
      the page. Repeat this step to produce another rectangle which is 2 squares wide and 1
      square deep, with the top left corner of the rectangle at the intersection of the first
      horizontal gridline from the top of the page and the fifth vertical gridline from the left
      edge of the page.

10.   Click on the text tool in the tool box, and drag a rectangle which is 7 squares wide by 1
      square deep, with the top left corner based on the top edge of the page at the first vertical
      gridline from the left. This time when you release the mouse button a text editing
      window appears. Enter the word 'Invoice' and press the Return key. The text 'Invoice'
      now appears in the center of the text box. Select a font, point size and attribute on the
      top tool bar of Arial 36 pt bold for the text.
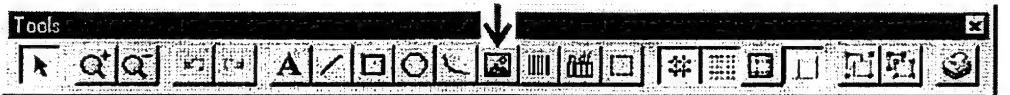
11.   Click on the graphic tool on the tool bar and drag a rectangle anywhere on the page that
      is one grid square in size. When you release the mouse button a file selector dialog box
      opens which prompts you to select a graphic file for entering in the box you have drawn.
      Select the file by browsing for *c:\program files\formscape2\examples\fslogo.bmp*, which
      is the FormScape logo as a Windows bitmap file. Click on Open, and the graphic now
      appears in the box on the page

12.   Your form should now contain two rectangles, a text ('Invoice') and a graphic showing
      the FormScape logo. Click on the small cross at the top right of the Form editing screen
      to return to the project tree. In the File menu at the top of the screen click on Save.

      Your blank form (i.e. the form you created in the above steps, but without any input data from
      the host system) should now look similar to the illustration on the next page. You can of course
      make a test print of the page if you wish by clicking on the Test Print button on the Graphic
      form editing tool bar, as previously done for Exercise 1.

Invoice
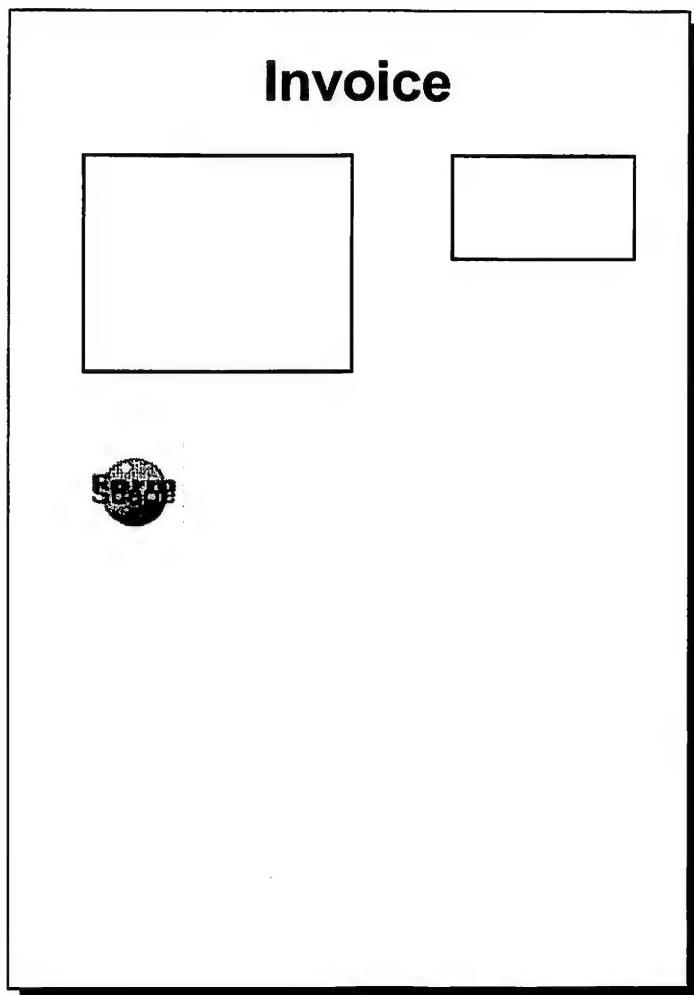
Sample form layout (Tutor1)

So far you have set up an input queue to FormScape, learnt about some of the objects and icons required, selected a printer and designed a form. Now it is necessary to fill in the form with the data generated by your host system. For the purposes of this tutorial, a sample of data has been included to develop this project. In a live situation you would capture the data using FormScape.

Carry out the following steps:

```
┌─────────────────────────────────────────────────────┐
│ Queue Process                                    ✕  │
├─────────────────────────────────────────────────────┤
│                                                       │
│   Samples                          ┌──────────┐     │
│                                    │    OK    │     │
│        ┌──┐   Open File            └──────────┘     │
│        │  │                         ┌──────────┐     │
│        └──┘                         │  Cancel  │     │
│        Copy a sample file into a PageBank          │
│                                                       │
│   Options                          ┌──────────┐     │
│                                    │Save Data │     │
│        ╱▔╲  Delete these PageBanks when a new job arrives: │
│       │    │  ☑ "Input"                             │
│        ╲__╱  ☐ "Process"                            │
│              ☐ All PageBanks                         │
│                                                       │
│        ┌─────────┐ ┌────────┐ ┌─────────┐           │
│        │Language..│ │Queue...│ │Action...│           │
│        └─────────┘ └────────┘ └─────────┘           │
└─────────────────────────────────────────────────────┘
```
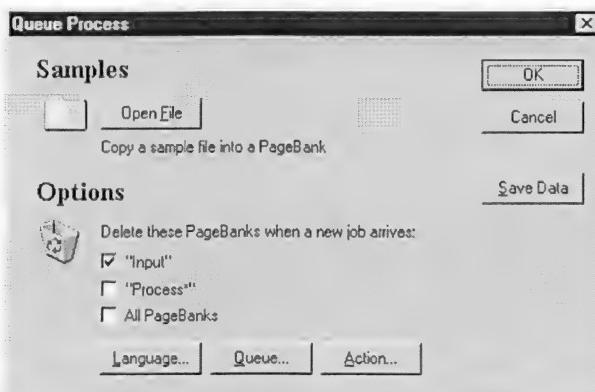
1.   Click on the Queue Process icon at the top of the project tree to display the Queue Process dialog. Ensure that the 'Input' option is selected. Click on the Open a file button. Use the Open Job File dialog box to browse and open the file *c:\program files\formscape2\examples\dem.txt*, the text file that contains the sample data. An Event Log window is displayed giving details about the job – 6 pages should have been loaded. Click on OK: you have now simulated a job arriving at the print queue. Click on OK again on the Queue Process dialog to close it.

2.   To fill out the form, it is necessary to be able to view the contents of the PageBank (the input pages of data) and the form layout you have created on the screen at the same time. To do this, click on the Reporter icon ('Pages') and click on the Map button in the Reporter dialog box. The screen now splits into two panes, with the input pages (Current Page) on the left and the blank Graphic form on the right (depending on the width of your FormScape display, the panes may be split vertically). The Current Page pane shows the first input page of data – click anywhere on it to make it active (the bar at the top of the pane changes color). At the top of the input page are four page arrows that allow you to 'flick' through the pages (in this example there are six pages). Clicking on these arrows causes the data on each page of the sample to be displayed.

```
┌──────────────────────────┐
│ S Brown                  │
│ 41 Crown Street          │
│ Botchington              │
│ Yorks                    │
│ LS1 9FR                  │
└──────────────────────────┘
```
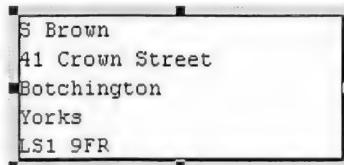
3.   Mark a rectangle around the address block on the Current Page by clicking and dragging, as before. This marks a block on the input page.

4.   The toolbar at the top of the screen also shows the position of the cursor on the input page, measured in columns (X) and lines (Y) of the text on the page. When you select a block to be mapped onto your form, a text box opens in the toolbar that you can edit to give your selected block a name (this is the name that will appear on the tree for the block). The toolbar also displays the size of the selected block in columns and lines.
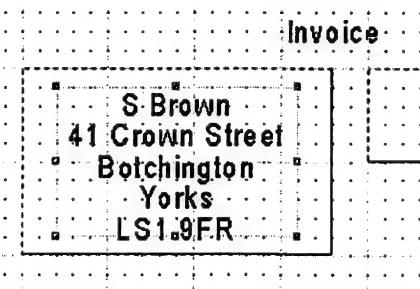
5. Use the **right** mouse button to drag this rectangle over to the Graphic form pane (the form layout you designed previously) and position it inside the larger of the two boxes you previously drew on the form. When you release the mouse button the text appears on the form.

6. The text is scaled according to the size of the block on the input page, and it is probably the wrong size for the moment. You can correct this by moving the cursor over one of the corners of the block so that it changes to a double-headed arrow, and then dragging the cursor to align it inside the box. You may also need to change the size of the text - with the text box still selected, choose a font and point size e.g. Arial 14 pt which allows the data to be displayed entirely within the box. Click on the Left justify button on the top toolbar and click on the Additional Tools button to select Top to align the text vertically in the box.

7. Repeat these steps using the date information on the input page and mapping it into the smaller box you previously created on the form.

8. To check your work, click on the Current Page pane, and using the page arrows at the top of the screen, flick through the input pages successively, noting that the text on Graphic form pane is updated each time. Check particularly that the address information is not cropped on some page(s) as a result of making the original mapped block too small or narrow. If this happens, place the cursor inside the defined block on the input page and click the mouse button to select the block. Move the cursor over one of the 'handles' of the block and drag the rectangle out to increase its size until the address information is correctly shown on the form for all the pages. If the box itself on the form proves to be too small for some pages, the text within it will wrap around automatically. If this happens, you can choose either to resize the box, or select a smaller font size to accommodate the text better within the box.

9. Close the Graphic form pane by clicking on the small cross in the top right hand corner. Do this again for the Current Page pane, so that you now have two tree views of the project. Remove one of these views (it doesn't matter which) by clicking on the double arrow at the top left of the view and select Remove Pane from the drop-down menu.

10. Click on the File menu and select Save to save your work. (A message may be displayed that asks if the PageBanks should be deleted. Tick the "Don't ask me this again" checkbox and click the "No" button).
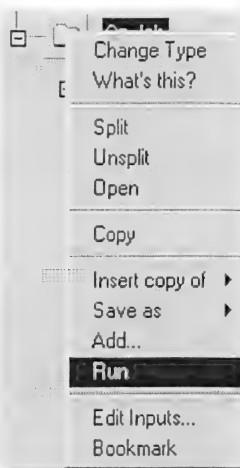
11. You can now print (run) the job (your project) by clicking on the On Job icon with the **right** mouse button and selecting Run from the drop-down menu. After a short pause, your printer will produce six pages of forms.

12. At any time you can display a preview of the entire job by clicking on the Print Graphic icon (currently labeled 'Top Copy'). A number of pages appear in preview on the screen as they would appear from the printer. As before, you can use the Page arrow buttons to flick through the pages, and zoom in or out on pages using the Zoom buttons. If, for instance, you repeatedly press the Zoom Out button you can eventually display a preview of all six pages of your project.

You have now created a project and a FormScape branch file 'Tutor1' by mapping prepared selected sample data onto a form you have designed. The next tutorial project (Tutor2) uses this project and enhances it by adding a second or File copy of the input pages to be produced at a different printer at the same time as the original set of pages.

3.2 TUTORIAL PROJECT 2

In this second project you will use the previous Project 1 you have just developed and enhance it by adding a second copy of the data, to be printed on a different printer.

Carry out the following steps: -

1. To start Developer, click on Start|Programs|FormScape2|Developer. The Developer logo appears, followed by a full-screen view of the project tree. Re-open the existing Tutor1 project and save it (to the Server) as a new project Tutor2 to make a copy of the project. Expand the tree using the + signs, as before.

2. Use the **right** mouse button to drag the 'Top Copy' Print Graphic icon down to the add-point below it. When you release the mouse button a drop-down menu appears. Select Copy. A new Print Graphic icon appears which is labeled Top Copy 1, and is an exact copy of the original. If you were now to run the application as it stands, you would get two identical sets of prints of the job from the same printer. Rename the new icon by double clicking its label and changing it to read 'File Copy'.

3. Expand the tree fully for the new 'File Copy' Print Graphic icon by clicking on the + signs to the left of each icon, until you reach a Text Box object labeled 'Text Box = Invoice'. Expand this branch again to produce a list of the properties (font, text attributes, color, point size) of this box:
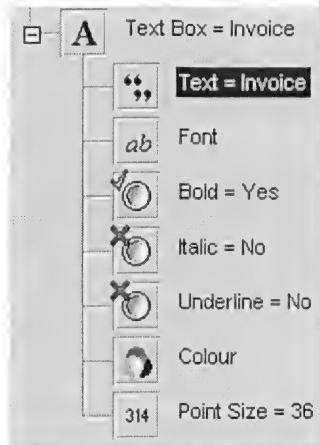
4. Double-click on the Text object labeled 'Text = Invoice' (click on the icon rather than the label) and change the text to read 'Copy Invoice'. Note that the Text Box object above this icon also changes its label to read 'Text Box = Copy Invoice'.

5. If you now click on the Form object for the File Copy branch you will see that the form is now headed 'Copy Invoice', which you have now changed to distinguish it from the form in the Invoice Top Copy branch (which is still labeled 'Invoice', as before).

6. Alternatively, steps 4 through 5 could have been carried out using the Graphic form Editor for the File Copy directly and editing the text box to read 'Copy Invoice' instead of 'Invoice'. The end result would be the same, but it is a useful exercise to verify that this is so, as well as demonstrating an alternative method.

You have now generated two sets of forms, a Top Copy set (headed 'Invoice') and a File Copy set (headed 'Copy Invoice'), which carry identical sample data.

7. Close the Graphic form editor to return to the project tree.

8. To change the printer for the File Copy set to a different destination printer, click on the printer icon for the File Copy branch and select an alternative printer from the available list.

9. Save the project by clicking on File|Save, and then verify that you have now developed two different forms (but with identical sample data) outputting to two different printers by clicking on the 'On Job' folder with the **right** mouse button and selecting Run. This will cause a set of six pages to be printed out at both printers. Alternatively, you can click on the File Copy Print Graphic icon with the **right** mouse button, select Run, and a set of six pages will be printed out from the File Copy printer only.

3.3   TUTORIAL PROJECT 3

In this third project you will use the previous project and enhance it by adding some conditional text to the set of pages i.e. text that appears on certain pages only of the sample data according to conditions established in the Developer module. In the exercise you will add the words 'CASH ONLY' to the Top Copy set, but only to those pages in the set of sample data that specify the payment method as CASH. The words 'CASH ONLY' are to be added as a large gray banner text diagonally across the entire form.

Carry out the following steps:

1.   To start Developer, click on Start|Programs|FormScape2|Developer. The Developer logo appears, followed by a full-screen view of the project tree. Re-open the existing Tutor 2 project and save it (to the Server) as a new project Tutor3 to make a copy of the project.

2.   Expand the tree using the + signs, as before, for the Top Copy branch, and click on the Form icon to display the Graphic form screen view.

3.   Add a Text Box by clicking on the text button in the tool bar, dragging a rectangle on the form that is six grid squares in each direction. Enter the text 'CASH ONLY' in the text box. Select a large font size e.g. Arial 60 pt bold and set center justification.

4.   Click on the Additional Tools button at the top left of the screen and select Rotation from the drop-down menu. Enter a rotation value of 45 degrees and click on OK. Note that the text 'CASH ONLY' is now rotated anti-clockwise by 45°.

5.   Select the color of the text to be gray by clicking on the color button and selecting a gray shade from the color palette. Click on OK.

You now have the text 'CASH ONLY' in a gray diagonal (45°) banner across the form, but it will of course appear on every page. You now need to modify the project so that this banner appears only on those pages that have the payment method as CASH, and not on those pages with the payment method as CREDIT. The sample data includes examples of both types, which you can see by flicking through the input pages using the Page arrows.

6.   Close the Graphic form by clicking on the small cross in the top right-hand corner of the pane.

7.   Expand the tree below the Form icon: note that a new branch has been added at the end of the tree labeled 'Text Box = CASH ONLY'.

8   Using the **right** mouse button, click the Text Box icon labeled 'Text Box3 = CASH ONLY' and select 'Change Type' from the drop-down menu.

If

9. A Change Object window appears: click on the Core category, and select the 'If' option. Click on OK. Note that the Text Box icon has now changed to an If icon. Click on the + sign to the left of this icon to expand the tree further.

```
 ⊟    If    Text Box3   If

            ⊗    Test = No     Yes/No

 ⊞    A    Yes = Cash Only    Text Box

            ⬧    No    Form
```
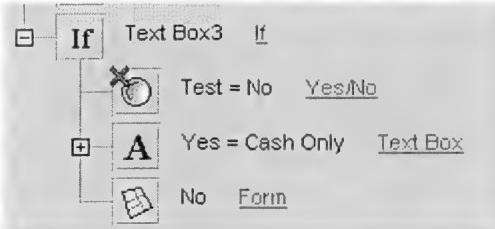
10. Click on the Reporter icon (currently labeled 'Pages') and on the Map button to produce a split screen with the input pages (Current Page) and the Graphic form. Close the Graphic form window and select Close to Tree. In the Current Page pane click on the Page arrows to find a page with the words 'Payment CASH'. Drag a rectangle around the text 'CASH'.

```
Ref    23/94031
Page 1
Payment  CASH
```
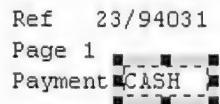
11. Check that the tree displayed in the Exploring window is expanded enough to show the If icon and the icons below it on its branch. Use the **right** mouse button to drag the 'CASH' rectangle across from the Current Page window to drop it on the Yes/No object labeled 'Yes = Cash Only' in the Exploring window. This object now changes to a Text Match object labeled 'Test = Yes'.

12. Flick through the input pages of sample data using the Page arrows and note how the icons in the 'If' branch change depending on the input page displayed. For an input page with CASH, 'Test = Yes' appears (icon shows a green tick), while for a sample input page with CREDIT, 'Test = No' appears instead (icon shows a red cross).

13. Verify that the banner 'CASH ONLY' now appears only on those pages with 'Payment CASH' text. This can be done by previewing the pages (click on the Print Graphic icon 'Top Copy' and flick through the pages with the Page arrows).

   Alternatively, you can print the job by clicking with the **right** mouse button on the 'On Job' icon and selecting Run from the drop-down menu which then appears. Two sets of forms will then be printed, but on some of the Top Copy set the additional text 'CASH ONLY' will be printed as a large diagonal banner.

14. Save the project by clicking on File|Save.

3.3 TUTORIAL PROJECT 4

As an extension of this project, you can now try using the Sentence object to join pieces of text together and place them on the page. This can then be formatted, even though the data changes from one page to the next. In this exercise, you will change the CASH ONLY text to include the 'Total' field from the input page.

Carry out the following steps:

1. To start Developer, click on Start|Programs|FormScape2|Developer. The Developer logo appears, followed by a full-screen view of the project tree. Re-open the existing Tutor3 project and save it (to the Server) as a new project Tutor4 to make a copy of the project.

4. Expand the tree for the Top Copy branch using the + signs, and click on the Form icon to enter the Graphic form editor view. Select the 'CASH ONLY' text box by clicking on it, and remove it from the page by right clicking on the text box and selecting Cut from the drop-down menu.

5. Display the Exploring tree view by clicking on the double-headed arrow and selecting Split Vertically. (Note that the If object has now disappeared – this was deleted when you deleted the 'CASH ONLY' box in step 2). Expand the tree using the + signs.

4. Click on the Graphic form editor pane, select the Text Box button from the toolbar and drag a rectangle on the page. Enter the text 'CASH ONLY' in the box that appears and press Enter to finish. Select a font and size of Arial 12 pt.

5. On the Exploring project tree, right-click on the Text object labeled 'Text = CASH ONLY' and select Change Type from the drop-down menu which then appears.

6. In the Change Object window, select the Sentence object in the Core category and click on OK. Note that the Text icon is now replaced by a Sentence icon:
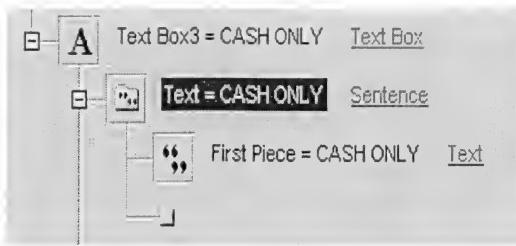
7. Click on the add-point at the end of the 'Text = CASH ONLY' branch and select a Text object from the Core category in the Create Object window. Click on OK. You have now added a new Text icon to the tree: click on it and enter the text 'Your invoice is for $' and click on OK.

8. Now click on the Text icon above it (labeled 'First Piece = CASH ONLY') and amend the text by adding a single space at the start of the text. Click on OK. (The reason for doing this will be apparent when the complete sentence is constructed in the correct order of its component items).

9. Open the page mapper by clicking on the Reporter icon (labeled 'Pages') and clicking on the Map button, to produce a split screen view. Click on the double-headed arrow at the top of the Graphic form editor and select Close to Tree.

10. Expand the tree until the add-point of the Sentence object appears. Using the scroll bars, position the Current Page pane so that the 'Total' field (bottom right-hand corner of the page) is visible, and drag a rectangle around the total value (make the rectangle wide enough to include larger numbers, if necessary).

1025.32

11. Use the **right** mouse button to drag this rectangle across to the tree in the right-hand view and drop it onto the add-point.

Text = CASH ONLYYour invoice is for $...

First Piece = CASH ONLY

Text = = Your invoice is for $

Object = \Data\Current Page\Block3
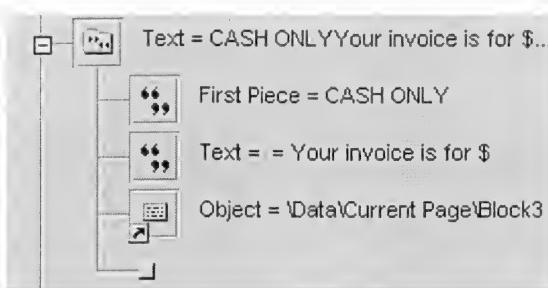
12. The sentence now has all its component parts, but the text will appear in the wrong order because of the way that you have assembled them on the tree. To correct this, click on the label of each object and use the positioning arrows to change the order of the objects around on the Sentence branch: -

Text = Your invoice is for $1025.32CA...

Text = = Your invoice is for $

Object = \Data\Current Page\Block3

First Piece = CASH ONLY
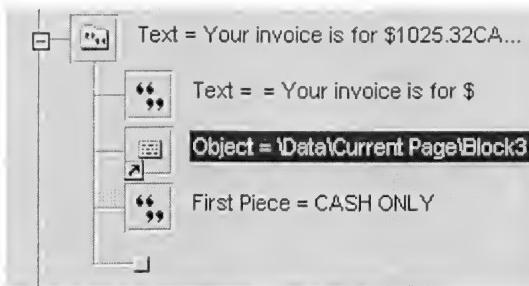
13. Verify by previewing the pages (click on the Top Copy Print Graphic icon). The text now reads 'Your invoice is for $XXXX CASH ONLY', where XXXX is the sum in the 'Total' field for each page.

14. Save the project by clicking on File|Save.

You can now test your project by carrying out the following steps:

1. Ensure that you save the project to the server by clicking on Save As|To the Server. A dialog box appears in which a folder with your computer name is displayed. Click twice on this folder to display two further sub-folders Actions and Processes. Select the Processes sub-folder and give the project the filename Tutor4. Click on OK to quit.

2. Close down Developer by clicking on File|Exit.

3. Check that a print queue for the project has been created by looking in your list of available printers: it should now include a printer called Tutor4.

4. To emulate printing your project on this printer, you need to copy the input data (the text file *dem.txt*) to the print queue, and this is most easily done using the Windows DOS Copy command. Firstly, copy the text file *c:\Program Files\FormScape2\Examples\dem.txt* to the root directory of your computer's hard disk (C:). Then, using the DOS Copy command, the command:

    copy c:\dem.txt \\[computer name]\Tutor4

    where [computer name] is the name by which your computer is recognized by your network. When you press Enter on the keyboard after the DOS command, the message '1 file(s) copied' appears, showing that the file has been sent to the printer. Alternatively, you can open the Tutor4 printer folder in your list of available printers so that it is visible on-screen alongside the DOS Command window. When you complete the DOS Copy command by pressing Enter on the keyboard, you can observe the Tutor4 print queue briefly displaying the Document Name sent to it before it is purged.

4 WORKED EXAMPLES

In this section are detailed a number of worked examples which are more complex than the tutorial exercises in Section 1 of this Guide (Getting Started). It is recommended that you work through them to familiarize yourself with the techniques involved for e.g. changing and moving objects on the Developer tree. Some of the examples require prepared sample data files which are to be found in your installed FormScape folder (normally *C:\Program Files\FormScape2/examples*).

The worked examples are:

1   Printing a banner page at the start of a print job.
2   Receiving different data in one Queue Process.
3   Making global changes.
4   Calculating a payment date.
5   Using the Page Splitter and Filter objects to position data.
6   Creating a summary report of all invoices printed throughout a day.
7   Determining the number of copies from the data.
8   Creating a report from comma-delimited data.
9   Adding headers and signatures to pre-formatted documents.
10   Inserting the correct contact details onto an invoice.

At the end is a section explaining what happens when you save a project, how to use branches to make global changes to your project and how to create a generic Queue Process. There is also a section explaining how to convert FormScape output to Adobe PDF format.

4.1 PRINTING A BANNER PAGE AT THE START OF A JOB

A banner page needs to be sent at the beginning of a job to identify who printed the document. A solution is to use a Binder object above the Form and Reporter objects and place details of the user and computer name on the banner page.
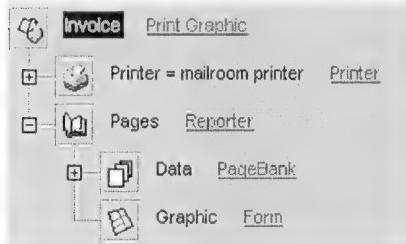
Expand the Print Graphic icon of the document that requires a banner page.

Change the Reporter to a Binder (found in the Forms category).

Note: the Reporter will not be deleted, but will be moved below the Binder.
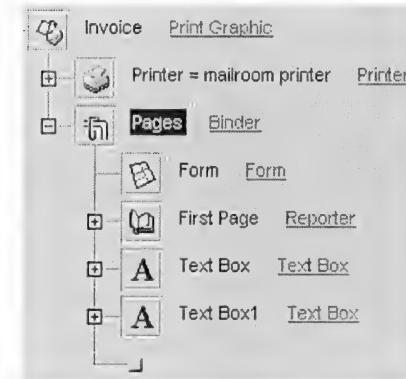
Add a Form object (found in the Form category) under the Binder object and move the Form object above the Reporter by using the blue up/down toolbar arrows in the main toolbar.

To display the User Name and Computer Name on the banner page, click on the add point and create two Text Box objects.

Change the Text objects to User Name and Computer Name (found in the Container tab).

4.1.1 CHANGING THE PAPER TRAY

The banner needs to be printed on green paper that is held in the Upper paper tray and the invoice needs to be printed on standard A4 white paper that is held in the default paper tray.

Right-click on the Banner page Form object and change it to a Page Setup object (found in the Printing category).
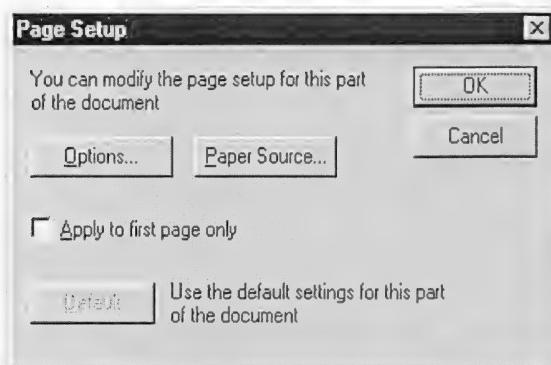
Click the Page Setup object icon to display the Page Setup dialog box. Click Paper Source and a list of available trays will be displayed.
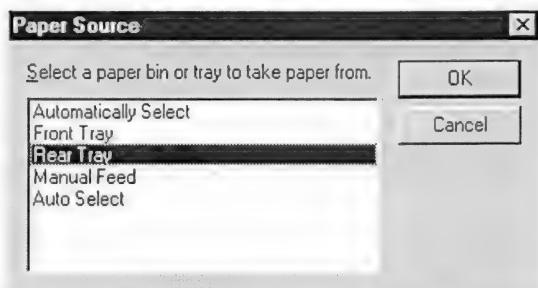
Select the required tray and click OK twice to return to the tree.

4.1.2 ADDING PAGE NUMBERS

We now want to add the text "Page 1 of x" to the Invoice, but the page number does not need to appear on the banner page.

First display the Graphic form by clicking on the Form icon and draw a Text Box where the page number should appear and press the Enter key. Right-click on the Text Box and select Explore.
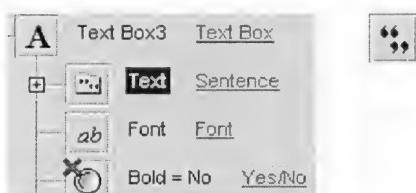
Now right-click on the Text object and change it to a Sentence object (found in the Core category).
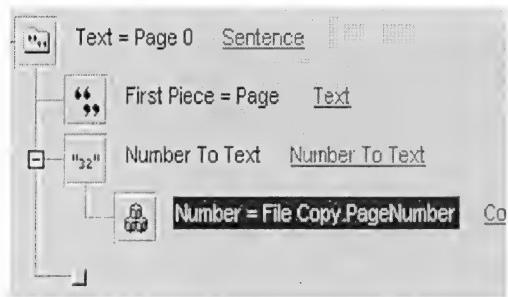
Expand the Sentence object, click on the Text object and enter the text "Page " with a space after the word.
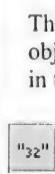
Then click on the add point under the Sentence object and select the Number To Text object (found in the Core category).
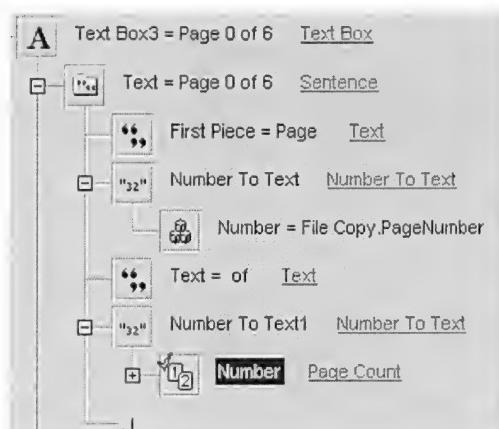
Expand the Number To Text object and change the Number object to a Page Number container (on the Container tab).

Click on the add point under the Sentence object and add a Text object. Click the Text object icon and enter the text " of " (be sure to add spaces before and after the word).

Click the Add point again and add another Number To Text object (from the Core category). Expand this and change the Number object to a Page Count object (found in the PageBank category).

Click on the Page Count object the Page Count dialog is displayed. Select 'input' from the drop-down list.

However, if you check the Print Preview you will see that the numbering starts at page 2. This is because the first page of the invoice is actually the second page that is printed. To display the correct numbering it is necessary to subtract 1 from the Page Number.

| A | Text Box23 = Page -1 of 6 |
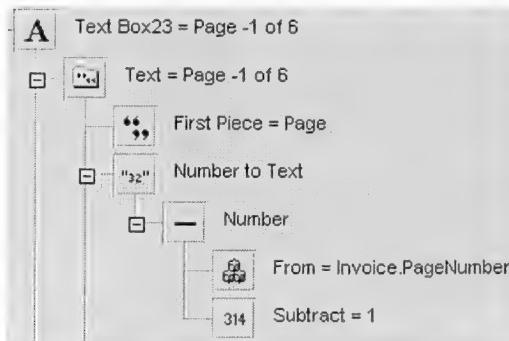
Change the Page Number property to a Subtract object (found in the Arithmetic category). The Page Number property will automatically be moved below. Click on the Number object labeled Subtract. Enter "1".

If you now check the Print Preview you should see that the banner page is displayed first (with no page number), and that the subsequent pages of the job are labeled "Page 1 of . . . ", etc.

4.2 RECEIVING DIFFERENT DATA IN ONE QUEUE PROCESS

Both Sales Orders and Pick Lists need to be sent to the same Queue Process and mapped onto their corresponding forms. We will use the Page Switch object to place the different types of data into different PageBanks. This example makes use of the file *sopick.txt* as sample data.
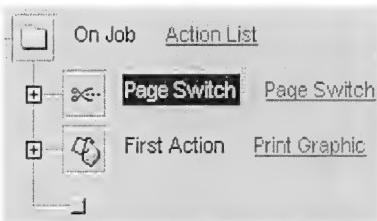
First click on the add point under the On Job folder and select Page Switch from the PageBanks category. Move this above the Print Graphic by clicking on its label and using the blue up arrow.
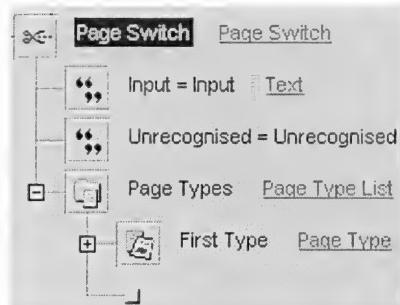
Click on the Page Switch icon to view the Page Switch dialog. The Input PageBank and the Default PageBank for unrecognized jobs are defined here, but in this example we can use the default values.

Expand the Page Switch object. The Page Type folder will contain a list of all the different Page Types that we want to test for.

We will set up the first Page Type object to test for Sales Orders. Click on the Page Type object.

Enter the text 'Sales Order PageBank' in the Output box. Then click the Map button.

The screen splits and the data is displayed in a Input Page pane along with the Exploring tree. Locate the text "S A L E S   O R D E R" on the Input Page and draw a box around it. Right-click and drag and drop this box onto the All of object. A test is automatically set up to check for the defined text in that position. Any pages that pass the test will be placed in the Sales Order PageBank.
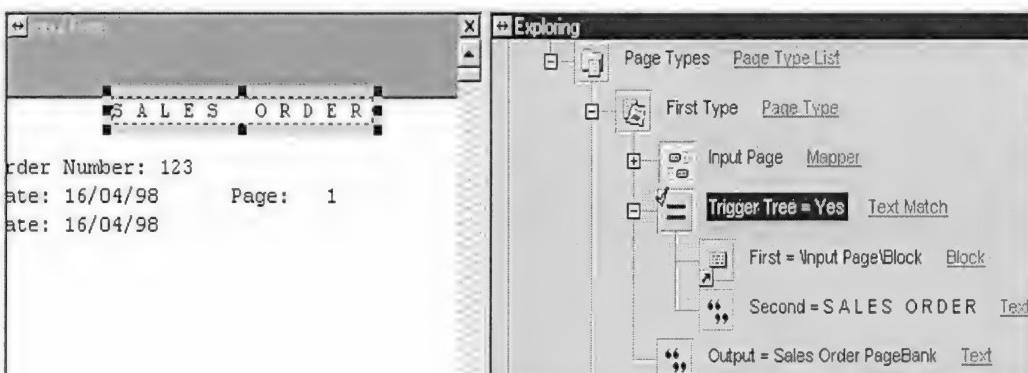
Close the Input Page pane and rename the first Page Type object labeled 'First Type' to 'Check for Sales Orders'.

Click on the add point under the Page Types object and select Page Type from the PageBanks category. Click on the new Page Type icon and enter "Picklist PageBank" in the Output field and then click the Map button to define the test.

In the Input Page pane, draw a box around the text "P I C K L I S T" and right click and drag and drop onto the All of object that you just created. All pages with PICKLIST in this position will now be placed in the Picklist PageBank.

Close the Input Page and rename the second Page Type "Check for Picklists".
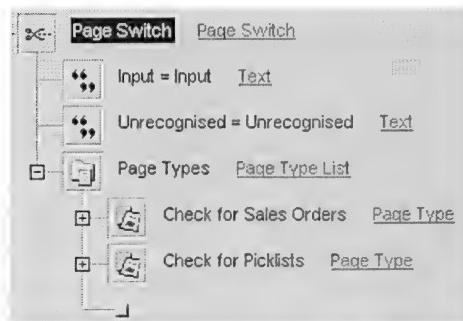
Right click on the Page Switch object and select Run. The new PageBanks will now be created – the results are listed in a Event Log window. Click on OK to close the Event Log window.

Now we need to create a Print Graphic for each type of document and to ensure that the data is taken from the correct PageBank. Click the minus sign next to the Page Switch object to collapse that part of the tree.
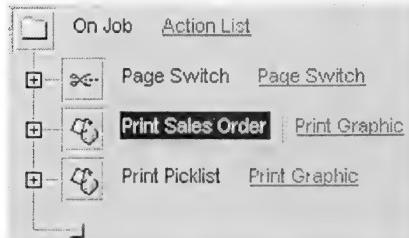
Click on the Add Point under the On Job folder. Add another Print Graphic (found in the Printing category). Rename one Print Graphic "Print Sales Order" and the other "Print Picklist".

Expand the Print Sales Order Print Graphic and the Reporter object underneath it.

Click on the PageBank object (labeled 'Data') and the Read PageBank dialog is displayed. Select "sales order pagebank" from the drop down list. Repeat this procedure for the Print Picklist Print Graphic to select "picklist pagebank".

The forms for both documents can now be created and the correct data will be mapped onto each form.

4.3   MAKING GLOBAL CHANGES

Some objects can be reused within a Queue Process or project. So that all the occurrences of these objects only need to be updated once, it may be desirable to store them in a single location. The following examples show how this can be achieved with the Memory object, the Read File object and the Dynamic Picture object.

4.3.1   USING THE MEMORY OBJECT TO MAKE GLOBAL CHANGES

In the following example we will make the FormScape logo and company telephone number available.

If we want the objects defined within the Memory object to be available to all objects under the On Job folder then we need the Memory object to be positioned above the On Job Folder.
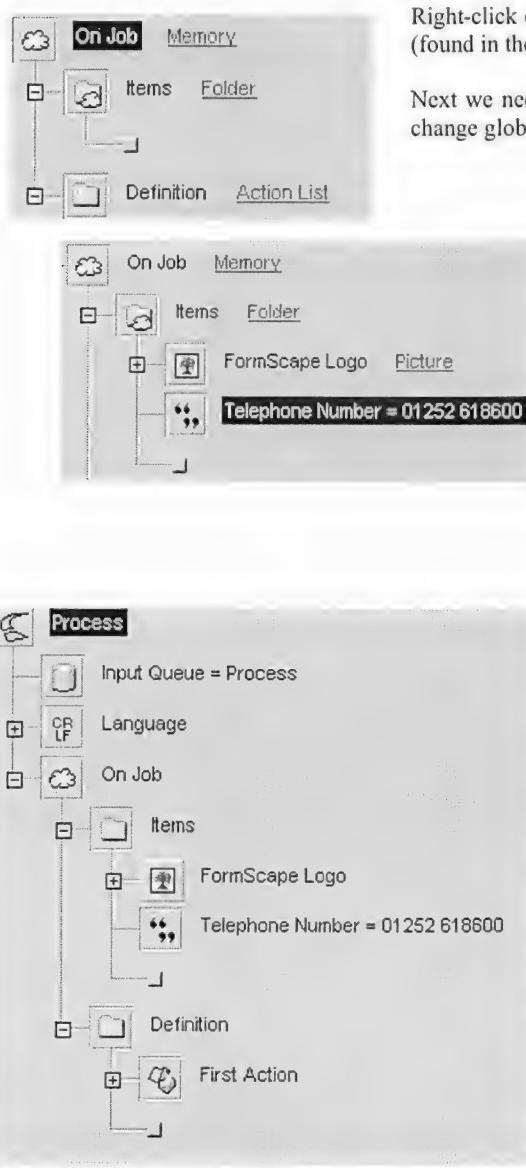
Right-click on the On Job object and change it to a Memory object (found in the Core category).

Next we need to add the objects that we would like to be able to change globally to the Items object.

Click on the add-point under the Folder object (labeled 'Items') and select Picture (found in the Forms category). You will be asked to browse for a picture; in this example we are using the FormScape splash screen (formscape/*splash.bmp*). Double-click on the label for the Picture object and enter the text "FormScape Logo".

Next click the add-point and select a Text object (found in the Core category). Click the Text object and enter the telephone number "01252 618600". Then double-click on the label for the Text object and enter "Telephone Number".

These objects will now be available on the Container tab.

If we expand the Action List object labeled 'Definition' we can use these objects in the Print Graphic.

To place the logo onto the form, expand the Print Graphic object, then expand the Reporter object.

Click on the Form object (labeled Graphic) to enter the Graphic form editor.

To insert the Picture onto the Form click the Object button from the tool bar.

Define the area on the form where you want the logo to appear. The Create Object dialog box will be displayed. Click on the Container tab.
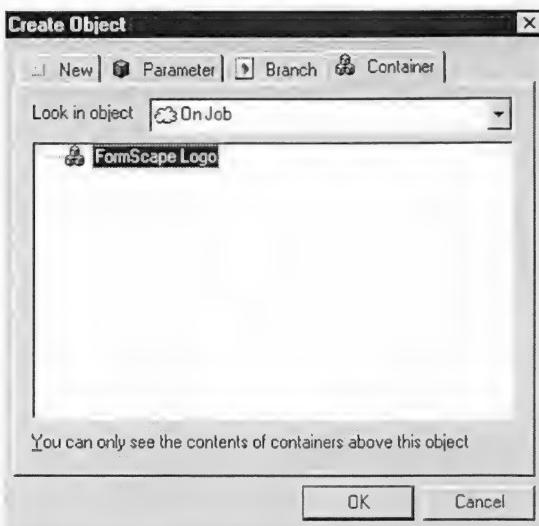
Under the On Job object, the FormScape Logo property is displayed. Select the FormScape Logo and click OK.

The FormScape Logo will now be displayed on the Graphic form editor in the area you previously defined.

Create Object dialog:

- New | Parameter | Branch | Container
- Look in object: On Job
- FormScape Logo
- You can only see the contents of containers above this object
- OK Cancel

To add the telephone number, click on the Text tool and draw a text box on the Graphic form. Press the Return key to leave the Text Box empty.

Text Box = 01252 618600   Text Box

Text = On Job.Telephone Number

Font   Font

Bold = No   Yes/No
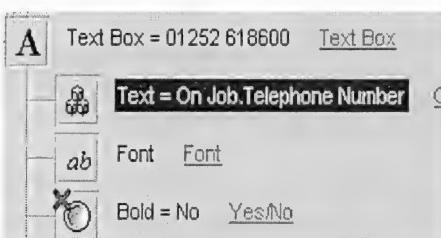
Right-click on the Text Box on the Graphic form and select Explore – the project tree is displayed in an Exploring pane.

Expand the Text Box object (under the Form object) and right-click on the Text object (labeled Text) – select Change Type. Select the Container tab and under the On Job category, select the Telephone Number and click OK.

The FormScape Logo and the Telephone Number can be re-used as many times as necessary. Try changing one of these under the Items list and the form will automatically be updated with the new picture or text.

4.3.2 USING THE DYNAMIC PICTURE OBJECT TO MAKE GLOBAL CHANGES

The Dynamic Picture object refers to an external location for the bitmap rather than holding the bitmap within FormScape.

To use the Dynamic Picture object, open the Graphic form editor (click on a Form object where you want the bitmap to appear). Select the Object tool from the Tool Bar and define the area for the bitmap – when you release the mouse button the Create Object dialog is displayed. Select the Dynamic Picture (found in the Forms category) and click OK. You will be asked to browse for the bitmap required. When you click on the Open button, the bitmap you selected is displayed.

Graphic   Form

Dynamic Picture   Dynamic Picture

Fixed Aspect = Yes   Yes/No

Original Size = No   Yes/No

Picture File = F:\formscape2.3-beta\Splash.bm.
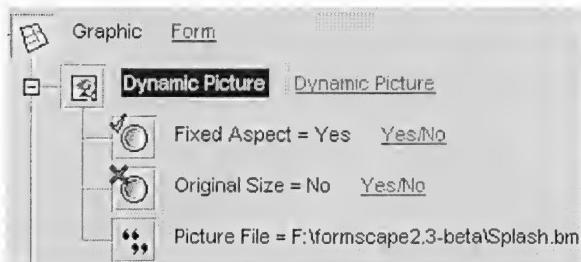
On the Graphic form editor, select the picture and right-click - select Explore from the pop-up menu. The project tree is displayed in an Exploring pane.

Expand the Dynamic Picture object on the tree. The path that is defined is held in the Text object below the Dynamic Picture object (note that the path and filename must exist on the FormScape server). If the picture needs to be updated, simply replace the bitmap file with a new bitmap file.

4.3.3 USING THE READ FILE OBJECT TO MAKE GLOBAL CHANGES

The Read File object allows you to use text that is held in an external file. Therefore, if the text needs to be changed, the file is updated and the changes are reflected within FormScape.

First define a text box on a form using the Text tool. Press the Return key to leave the Text Box empty. Right click on the Text Box on the form and select Explore.

Expand the new Text Box object and change the Text object underneath it to a Read File object (found in the Files category).
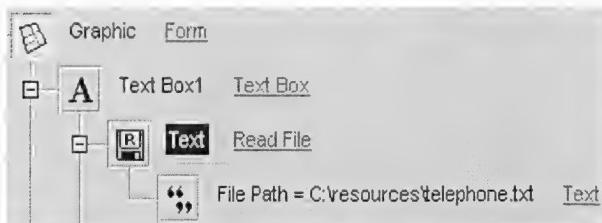
Click on the Read File object and you can browse for the required file. In this example the file that holds the telephone number is C:\resources\telephone.txt.

From now on, when the text needs to be changed, the file is updated and this change is reflected wherever the file is referred to within the project.

4.4 CALCULATING A PAYMENT DATE

The date that the customer is due to pay the invoice by is 28 days after the invoice is printed. This needs to be stated on the invoice form.

First click on the Form object to display the Graphic form editor. Create a new Text Box on the form and enter the text "Payment should be made by " (be sure to leave a space after the text).
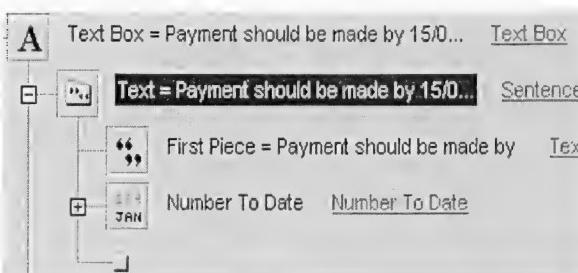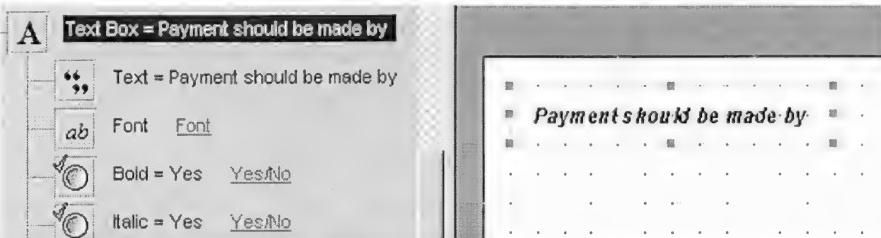
Right click on the Text Box on the form and select Explore – the new Text Box object has been added under the Form object. Click on the plus sign to expand the Text Box object.

Change the Text object (under the Text Box object) to a Sentence object to allow the date to be added – right-click on the object, select Change Type and select Sentence from under the Core category.

Expand the Sentence object and click on the add point. Select the Number to Date object (found in the Core category).

Next, expand the Number to Date object.

Change the Number object below it to an Add object (found in the Core category).
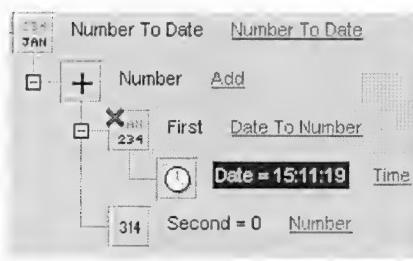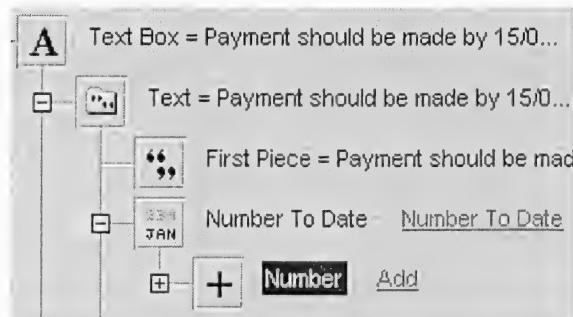
The arithmetic calculation needs to add 28 days to the current date. This is achieved by converting the date to a number, adding 28 and then converting the calculated value back into a date.

Expand the Add object and change the 'First' Number object to a Date to Number object (found in the Core category).

Expand the Date to Number object and change the Text object below to a Time object (also found in the Core category).

By default the Time object will display the current time, but we require the current date.

Click on the Date To Number object to display the Date To Number dialog, which shows the current date formats. The Time object should use the same format so that the value is recognized as a valid date.

Date To Number dialog:

Select the international setting for the current date input:

English (United Kingdom) ▼

Short date sample:

25/08/98

Long date sample:

25 August 1998

Note: if the year part of the current date input is 2 digits, then values between 50 and 99 will translate to 1950 to 1999, and values between 00 and 49 will translate to 2000 to 2049.

OK   Cancel

Time dialog:

OK   Cancel

International
- ⦿ System Setting   English (United Kingdom)
- ○ User Setting   English (United Kingdom)
- ○ Other Setting   English (United Kingdom) ▼

Format
- Date and time (15/04/00 15:16:38)
- Long date and time (15 April 2000 15:16:38)
- **Date (15/04/00)**
- Long date (15 April 2000)
- Time (15:16:38)
- Abbreviated weekday name (Sat)
- Full weekday name (Saturday)
- Abbreviated month name (Apr)

Tip: you can create your own new formats with the sentence object.

Click the Time object to display the Date/Time dialog. Select the correct date format from the list and click on the OK button.

To finish the calculation we need to add 28 to the date value. Click on the Number object labeled 'Second = 0' and enter 28. The payment date (i.e. 28 days after the current date) is now displayed on the form.

Tree view:

- **A** Text Box = Payment should be made by 13/0...   Text Box
  - Text = Payment should be made by 13/0...   Sentence
    - " First Piece = Payment should be made by   Text
    - Number To Date   Number To Date
      - **+** Number   Add
        - First   Date To Number
          - ○ Date = 15/04/00   Time
        - 314 **Second = 28**   Number

Form: *Payment should be made by 13/05/00*

- Number To Date   Number T
  - **+** Number   Add
    - First   Date To N

If you want to display the date in a different format, click on the Number to Date object (this is the object that appears above the Add object on the tree). You can select a different date format from this editor by selecting a date style or language setting.

4.5 USING THE SPLITTER AND FILTER OBJECTS TO POSITION DATA

The data below (taken from the sample data file *salesorder.txt*) shows a Sales Order that needs to be mapped onto a form in FormScape. There are two problems with the data that need to be addressed. First, the 'Ship to' information does not always appear, and so the Purchase Order number and following data is not in a fixed position. Second, the dotted lines (---------) should not appear on the form.

4.5.1 USING THE SPLITTER FOR MOVING DATA

Load the file (click on the Queue Process icon, click on the Open a file button and select the salesorder.txt file).
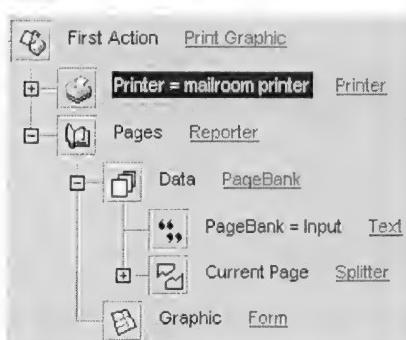
Expand the Print Graphic object , and expand the Reporter and PageBank objects.

Change the Mapper object labeled Current Page to a Splitter object (found in the PageBanks category).

Click on the Splitter object icon to open the Current Page editor in a new pane.

On this pane we need to define which text decides where to split the page and the relative position where the page should be split.

In this example we shall be looking for the text "Purchase Order" and will be splitting the page directly above. To split the page, drag the horizontal red dotted line down from the top edge of the mapped page so that it is exactly above the line containing the text "Purchase Order".
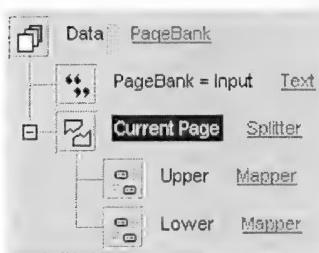
If we expand the Splitter object in the Exploring tree, we see that the page is now divided into two Mappers.

The Upper Mapper will display all the information up to the line above 'Purchase Order' and the Lower Mapper will display everything below.

```
Data  PageBank
      PageBank = Input   Text
      Current Page   Splitter
         Upper   Mapper
         Lower   Mapper
```

```
       Purchase Order: 12345
       Credit Terms: 15

Dpt Item Number        Due
-----------------------------------
   1 p10987            17/10/98
```

Now "Purchase Order" will always appear at the top of the Lower Mapper regardless of the length of the details in the Upper Mapper, so the Purchase Order can easily be mapped onto the form.

4.5.2 USING THE LINE FILTER TO REMOVE UNWANTED DATA

Next we need to remove the unwanted dotted lines from the data.

Locate the Lower Mapper and change it to a Line Filter (found in the Pages category). Split the Exploring pane (click on the double arrow and select Split Horizontal or Split Vertical).

Expand the Line Filter object in both panes. Click on the Mapper object labeled 'Current Line' in the left or top pane.

```
       Purchase Order: 12345
       Credit Terms: 15

Dpt Item Number        Due
-----------------------------------
   1 p10987            17/
-----------------------------------
```

The data is displayed with a white band on the first line of data. Move the cursor above the white line (the cursor changes to a double arrow), left-click and drag the line down to the first dotted line.

```
Current Page   Splitter
   Upper   Mapper
   Lower   Line Filter
      Current Line   Mapper
      For Each Line = Yes   Text Match
         First = \Current Line\Block   Block
         Second   Text
      Passed   Mapper
      Failed   Mapper
      Insert Blanks = Yes   Yes/No
```
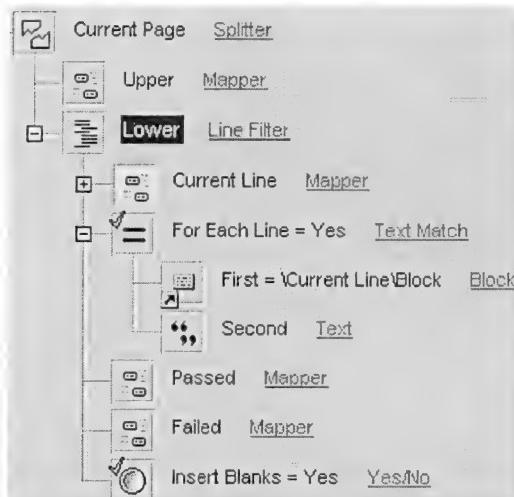
Draw a box around the first "-" and drag and drop with the right mouse button onto the All Of object.

Now all the dotted lines will be held in the Passed Mapper so we only need to map the contents of the Failed Mapper onto the form.

4.6 CREATING A SUMMARY REPORT OF INVOICES PRINTED THROUGHOUT THE DAY

A report needs to be printed at 23:00 every day to give details of all the invoices processed throughout the day. A Queue Process is required to collect the data and a Scheduler is also required to print the report at 23:00. The data is taken from the sample file *Invoice.txt* found in the FormScape/examples directory. In this example we will assume that a Print Graphic called Invoices has already been created using this sample file.

4.6.1 ADDING THE SCHEDULER

Both the Queue Process and the Scheduler need to run within the same thread, so they need to be defined under the same Process.

Start a new project.

Right click on the Queue Process object and change it to a Process List object.

Click on the add-point and add a Queue Process object.

Click on the Add point again and select a Scheduler object (Both these objects can be found in the Core category).
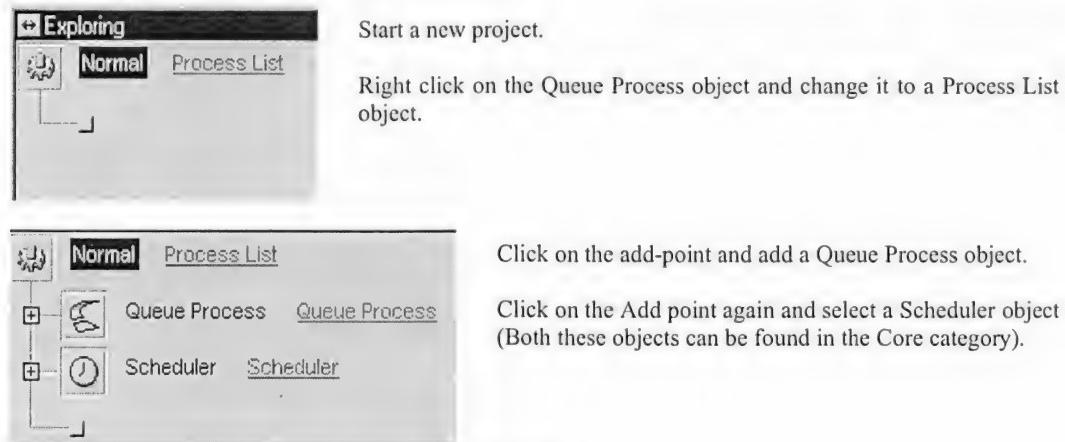
4.6.2 COLLECTING THE SUMMARY DATA

The Queue Process now needs to be defined to collect the summary data. There will probably be a Print Graphic that prints the invoices as the jobs arrive at the print queue.

Click on the add-point below the On Job object in the Queue Process branch, and add a Scan object (found in the Core category).

Expand the Scan object.

Replace the Print Graphic object labeled 'First Action' with an Add Page object (found in the PageBanks category).

Expand the Add Page object.

Replace the Text object labeled 'Input' with a Builder object (found in the PageBanks category).

Click the Add Page object and enter the name of the PageBank that will hold the summary data ("Invoice Summary" in this example). Click OK.

Click the PageBank object (below the Scan object) and ensure that the data will be taken from the default PageBank 'Input'. Click OK.

Click on the Scan object and click on the Map button to display the input pages.

Click on the Builder object to display the new data page. Define the areas of data required in the original data and using the right mouse button, drag and drop each block onto the new data page. This example uses the Customer Name, Invoice Number and Total.
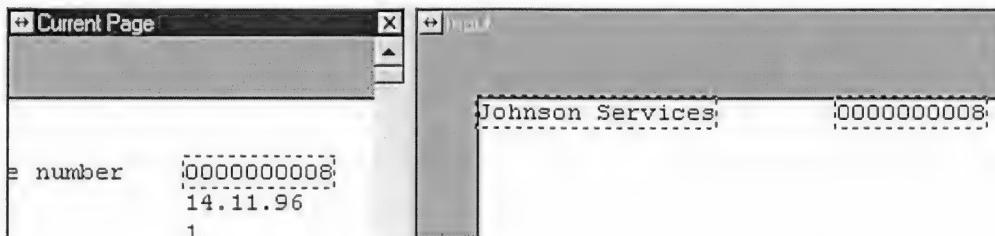
Because the Autodelete flag is not set, all invoices processed throughout the day will be held in the 'Invoice Summary' PageBank.

Close the Input pane and Current Page pane to return to the project tree. Right-click on the Scan object and select Run to create the PageBank "Invoice Summary".

4.6.3 CREATING THE REPORT

The report needs to be set up so that it is printed at 23:00, using the Scheduler object. Click on the Scheduler object and set the time of day to run as 23:00.
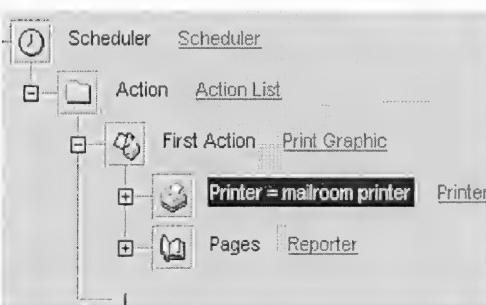
Expand the Scheduler object and the Print Graphic object below it.

First Action   Print Graphic
  Printer = mailroom printer   Printer
  Pages   Reporter

Change the Reporter object (labeled 'Pages') to a Form object (found in the Forms category).

First Action   Print Graphic
  Printer = mailroom printer   Printer
  Pages   Form
    Layout   Layout

Click on the Form object to display the Pages editor.

Click the Object tool on the toolbar and draw a box that covers most of the page (leaving space at the top of the page for headings).

Select the Layout object from the Forms category

First Action   Print Graphic
  Printer = mailroom printer   Printer
  Pages   Form
    Layout   Layout
      Graphic   Reporter
        Data   PageBank
        Graphic   Binder

Close the Pages form to return to the tree.

Expand the Form object and expand the Layout object underneath it.

Change the Binder object below it into a Reporter object (found in the Forms category).

Expand the Reporter object.

First Action   Print Graphic
  Printer = mailroom printer   Printer
  Pages   Form
    Layout   Layout
      Graphic   Reporter
        Data   PageBank
        Graphic   Form

Change the Binder object below the Reporter object to a Form object (found in the Forms category).

The data for the report needs to be taken from the "Invoice Summary" PageBank.

Click on the PageBank object (labeled 'Data') and select 'Invoice summary' from the drop down list.

Click on the Reporter object and click on the Map button to map across the required data to the form.

Next, shrink the page size to fit the data. With the Graphic from editor selected, select Options from the View menu, click the Paper Size tab and click on the Shrink to fit button. Clear the Auto-size check box.

Preferences ☒

Measurement | Paper Size | Background | Zoom | Font Sizing |

The default paper size is set by the printer object, but you can change it by dragging the edges of the form.

[Default]

Set the form size to the default paper size:
A4 Sheet, 210- by 297-millimeters

[Shrink to fit]

Shrink the form to enclose the objects currently drawn on it.

☐ Auto Size

When you open the form for editing or change the printer settings, it will be made the same size as a whole page.

[OK] [Cancel] [Apply]

Return to the project tree and click on the first Form object (labeled 'Pages') under the Scheduler object. If you now click twice on the Object block on the form (to refresh it) it will now show the data from each page in the PageBank. Add title and headings as required.

Invoice Summary Report

Customer Name	Invoice Number	Total
AB Jones Dairy	0000000001	1279.00
Bechlone Trail Holidays	0000000002	19345.95
Cartwright Services	0000000003	91.10
Ice Frozen Foods	0000000022	1279.00

4.6.4 EMPTYING THE PAGEBANK

Once the Invoice Summary Report has been printed the PageBank should be emptied before the new invoices are printed the following day.

Click the add point below the Action object and select Empty PageBank (from the PageBank category).

Click on the Empty PageBank object and select the "Invoice Summary" PageBank from the drop down list.

4.7 DETERMINING THE NUMBER OF COPIES REQUIRED FROM THE DATA

The customer number determines the number of copies required for an invoice. If the customer number starts with 'I' there should be 2 copies, 'F' 3 copies and 'R' 4 copies. This can be achieved by retrieving the required letter from the data and using a Scan object and a Range object to create the correct number of copies. In this example we will assume that a Print Graphic called Invoice has already been created using the sample text file *Invoice.txt*.

4.7.1 RETRIEVING THE LETTER FROM THE DATA

We need to make the letter available to other objects on the tree, so first change the On Job object to a Memory object (found in the Core category).

Expand the Action List object labeled 'Definition' and add a Scan object (found in the Core category). Move the Scan object above the Print Graphic object using the blue up arrow.

Click on the add point below the Items folder and select a Text object from the Core category. Double click on the label of this object and rename it 'Code'.

Click on the Scan object and click on the Map button. The window is divided into two panes. The data is displayed on the Current Page pane and the tree in the Exploring pane.

In the project tree, expand the Memory object and the Scan object.

Change the Print Graphic object (below the Scan object) to a Copy Text object (found in the Core category). Expand the Copy Text object.

On the Current Page pane, draw a box around the first character of the customer number. Right-click and drag and drop onto the Text object under the Copy Text object labeled 'From'.

Change the Text object below it labeled 'To' to the Container named Code (on the Container tab). The resulting tree is shown below: -

On Job   Memory
  ⊟ Items   Folder
          Code   Text

  ⊟ Definition   Action List
      ⊟ Scan   Scan
          ⊞ Data   PageBank
          ⊟ Action   Action List
              ⊟ First Action   Copy Text
                      From = \Data\Current Page\Block
                      To = On Job.Code   Container Sh

  ⊞ **Invoice**   Print Graphic

Right click on the Scan object and select Run. The block value will now be copied into the Code property.

Next, change the Print Graphic object labeled 'Invoice' to a Scan object (found in the Core category).

(The Print Graphic object will not be deleted, but it will be moved to the Action section of the Scan object).

Expand the Scan object you just created.

Change the PageBank object labeled 'Data' to a Range object (found in the Core category).

Invoice   Scan
  ⊟ Data   Range
          314   Start = 1   Number
          314   Step = 1   Number
          314   End = 0   Number
          314   Current = 0   Number
  ⊞ Action   Print Graphic

Expand the Range object. The Start value should be set to 1, the Step value should be set to 1 – click on the icons to change the values of these objects.

The End number depends on the customer number code from the data. We shall use a Lookup object to select the correct value for the End number.

Change the Number object labeled 'End' to a Text To Number object (found in the Core category). Expand this object and change the Text object below it to a Lookup object (found in the Deprecated category).

⊟ 21   End   Text To Number
    ⊟   Text   Lookup
        ⊟ a,b   Table   Tokenizer
                Text = I=2,F=3,R=4   Text
                Delimiter = ,   Text
                Key = On Job.Code   Container Shortcut

Expand the Lookup object.

Next we need to define the table (Tokenizer) in the Lookup object. Expand the Tokenizer object and click on the Text object labeled 'Text.' Enter "I=2,F=3,R=4". *Note that this is case sensitive.*

Click on the Text object below it labeled 'Delimiter' and enter ",".

Finally, change the Text object labeled 'Key' to the Code Container (on the Container tab). The correct number of invoice copies will now be printed.

Note: This example assumes that each print job only contains one invoice.

4.8 CREATING A REPORT FROM COMMA-DELIMITED DATA

The data that we require for a report is in the sample data file *speedcomma.txt* and is comma-delimited, i.e. each record appears on one line and the fields are separated by commas, as shown below:

The report is divided into three sections; the first details the name, date and penalty for a speeding offense. The second details how the offense was recorded, town, speed and speed limit. The final section gives some summary information.

4.8.1 PUTTING THE DATA INTO A BAG

With the Delimited Data Parser and other Container objects, we can manage data in this format more easily. Instead of the incoming data being placed into a PageBank, we are going to manage this data via a bag.

To stream the incoming comma separated data into a bag, change the ASCII Language object to a Delimited Data Parser object (found in the Containers category). Expand this object to display its child object.

Click on the icon to check that the configuration is correct for our data – the Delimited Data Parser dialog is displayed.

Click on the Sample Data button to load the sample data *speedcomma.txt* from the FormScape/Examples directory. The default values for the Delimiter Character, Record Termination and Qualifier Character options can be used for this example.

The data does not include any headings to identify the meaning

Change the Reporter object (labeled 'Pages') to a Form object (found in the Forms category).

```
├─ 🐾 First Action    Print Graphic
│   ├─ ⊞ 🖨 Printer = mailroom printer   Printer
│   └─ ⊞ 📖 Pages    Reporter
│   ┘
```

Click on the Form object to display the Pages editor.

Click the Object tool on the toolbar and draw a box that covers most of the page (leaving space at the top of the page for headings).

Select the Layout object from the Forms category

```
🐾 First Action    Print Graphic
├─ ⊞ 🖨 Printer = mailroom printer   Printer
└─ ⊟ 📖 Pages    Form
        └─ ⊞ 🔲 Layout    Layout
```

Close the Pages form to return to the tree.

Expand the Form object and expand the Layout object underneath it.

Change the Binder object below it into a Reporter object (found in the Forms category).

Expand the Reporter object.

```
🐾 First Action    Print Graphic
├─ ⊞ 🖨 Printer = mailroom printer   Printer
└─ ⊟ 📖 Pages    Form
        └─ ⊟ 🔲 Layout    Layout
                └─ ⊟ 📖 Graphic    Reporter
                        ├─ ⊞ 🗐 Data    PageBank
                        └─ ⊞ 🔲 Graphic    Binder
```

Change the Binder object below the Reporter object to a Form object (found in the Forms category).

The data for the report needs to be taken from the "Invoice Summary" PageBank.

Click on the PageBank object (labeled 'Data') and select 'Invoice summary' from the drop down list.

```
🐾 First Action    Print Graphic
├─ ⊞ 🖨 Printer = mailroom printer   Printer
└─ ⊟ 📖 Pages    Form
        └─ ⊟ 🔲 Layout    Layout
                └─ ⊟ 📖 Graphic    Reporter
                        ├─ ⊞ 🗐 Data    PageBank
                        └─ 📄 Graphic    Form
```

Click on the Reporter object and click on the Map button to map across the required data to the form.

Next, shrink the page size to fit the data. With the Graphic from editor selected, select Options from the View menu, click the Paper Size tab and click on the Shrink to fit button. Clear the Auto-size check box.

Preferences ☒

Measurement | Paper Size | Background | Zoom | Font Sizing

The default paper size is set by the printer object, but you can change it by dragging the edges of the form.

Default

Set the form size to the default paper size:
A4 Sheet, 210- by 297-millimeters

Shrink to fit

Shrink the form to enclose the objects currently drawn on it.

☐ Auto Size

When you open the form for editing or change the printer settings, it will be made the same size as a whole page.

OK | Cancel | Apply

Return to the project tree and click on the first Form object (labeled 'Pages') under the Scheduler object. If you now click twice on the Object block on the form (to refresh it) it will now show the data from each page in the PageBank. Add title and headings as required.

Invoice Summary Report

Customer Name	Invoice Number	Total
AB Jones Dairy	0000000001	1279.00
Bechlone Trail Holidays	0000000002	19345.95
Cartwright Services	0000000003	91.10
Ice Frozen Foods	0000000022	1279.00

4.6.4 EMPTYING THE PAGEBANK

Once the Invoice Summary Report has been printed the PageBank should be emptied before the new invoices are printed the following day.

Click the add point below the Action object and select Empty PageBank (from the PageBank category).

Click on the Empty PageBank object and select the "Invoice Summary" PageBank from the drop down list.

4.7 DETERMINING THE NUMBER OF COPIES REQUIRED FROM THE DATA

The customer number determines the number of copies required for an invoice. If the customer number starts with 'I' there should be 2 copies, 'F' 3 copies and 'R' 4 copies. This can be achieved by retrieving the required letter from the data and using a Scan object and a Range object to create the correct number of copies. In this example we will assume that a Print Graphic called Invoice has already been created using the sample text file *Invoice.txt*.

4.7.1 RETRIEVING THE LETTER FROM THE DATA

We need to make the letter available to other objects on the tree, so first change the On Job object to a Memory object (found in the Core category).

Expand the Action List object labeled 'Definition' and add a Scan object (found in the Core category). Move the Scan object above the Print Graphic object using the blue up arrow.

Click on the add point below the Items folder and select a Text object from the Core category. Double click on the label of this object and rename it 'Code'.

Click on the Scan object and click on the Map button. The window is divided into two panes. The data is displayed on the Current Page pane and the tree in the Exploring pane.

In the project tree, expand the Memory object and the Scan object.

Change the Print Graphic object (below the Scan object) to a Copy Text object (found in the Core category). Expand the Copy Text object.

On the Current Page pane, draw a box around the first character of the customer number. Right-click and drag and drop onto the Text object under the Copy Text object labeled 'From'.

```
invoice number       0000000001
date                 14.11.96
page                 1
order number         7PO456
customer number      F 1654
```

Change the Text object below it labeled 'To' to the Container named Code (on the Container tab). The resulting tree is shown below: -

Right click on the Scan object and select Run. The block value will now be copied into the Code property.

Next, change the Print Graphic object labeled 'Invoice' to a Scan object (found in the Core category).

(The Print Graphic object will not be deleted, but it will be moved to the Action section of the Scan object).

Expand the Scan object you just created.

Change the PageBank object labeled 'Data' to a Range object (found in the Core category).

Expand the Range object. The Start value should be set to 1, the Step value should be set to 1 – click on the icons to change the values of these objects.

The End number depends on the customer number code from the data. We shall use a Lookup object to select the correct value for the End number.

Change the Number object labeled 'End' to a Text To Number object (found in the Core category). Expand this object and change the Text object below it to a Lookup object (found in the Deprecated category).

Expand the Lookup object.

Next we need to define the table (Tokenizer) in the Lookup object. Expand the Tokenizer object and click on the Text object labeled 'Text.' Enter "I=2,F=3,R=4". *Note that this is case sensitive.*

Click on the Text object below it labeled 'Delimiter' and enter ",".

Finally, change the Text object labeled 'Key' to the Code Container (on the Container tab). The correct number of invoice copies will now be printed.

Note: This example assumes that each print job only contains one invoice.

4.8 CREATING A REPORT FROM COMMA-DELIMITED DATA

The data that we require for a report is in the sample data file *speedcomma.txt* and is comma-delimited, i.e. each record appears on one line and the fields are separated by commas, as shown below:

```
21/06/98,camera,High Street,Aldershot,Alex,Wright,43,3
30/06/98,camera,A31,Alton,Stuart,Malt,90,70,3 penalty
03/07/98,officer,A30,Basingstoke,Karen,Fuller,85,70,6
```

The report is divided into three sections; the first details the name, date and penalty for a speeding offense. The second details how the offense was recorded, town, speed and speed limit. The final section gives some summary information.

4.8.1 PUTTING THE DATA INTO A BAG

With the Delimited Data Parser and other Container objects, we can manage data in this format more easily. Instead of the incoming data being placed into a PageBank, we are going to manage this data via a bag.

To stream the incoming comma separated data into a bag, change the ASCII Language object to a Delimited Data Parser object (found in the Containers category). Expand this object to display its child object.

Click on the icon to check that the configuration is correct for our data – the Delimited Data Parser dialog is displayed.

Click on the Sample Data button to load the sample data *speedcomma.txt* from the FormScape/Examples directory. The default values for the Delimiter Character, Record Termination and Qualifier Character options can be used for this example.

The data does not include any headings to identify the meaning of each field (column of data). By default, each column is named Field1, Field2, and so on. It's good practice to change the headings to something more intelligible. The benefits of this will become apparent when mapping data later. Click in the column in the table of data and changing its name in the Fieldname text box.

Rename:
Field1 to *Date*
Field2 to *Method*
Field3 to *Road*
Field4 to *Town*
Field5 to *Forename*
Field6 to *Surname*
Field7 to *Speed*
Field8 to *Speed Limit*
Field9 to *Penalty*

When the changes are made, click on OK to close the dialog.

To fill the bag using the current configuration, click on the Queue Process object and click on the Open a file button to import *speedcomma.txt*.

The Delimited Data Parser child Bag object is now filled and can be viewed by clicking on its icon.

```
□ ⏲ Root
   □ 🗀 RECORDSET
      □ 🗀 RECORD
            ❞ Date : 21/06/98
            ❞ Method : camera
            ❞ Road : High Street
            ❞ Town : Aldershot
            ❞ Forename : Alex
            ❞ Surname : Wright
            ❞ Speed : 43
            ❞ Speed Limit : 30
            ❞ Penalty : 3 penalty points £40 fine
      ⊞ 🗀 RECORD
      ⊞ 🗀 RECORD
      ⊞ 🗀 RECORD
      ⊞ 🗀 RECORD
```
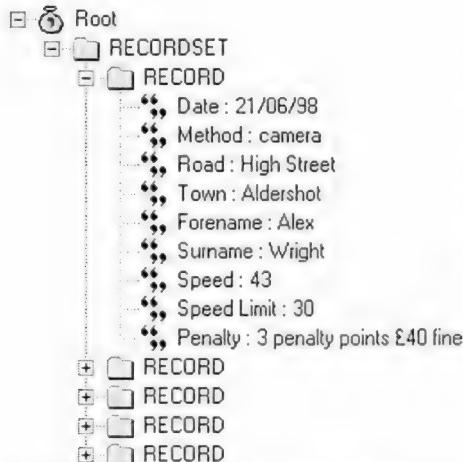
The bag should now look like the illustration opposite. There are 5 records. The first record has been expanded so that the child field names and text values can be read.

The bag is exposed by the root Queue Process object, so when it is needed later in the project branch, rather than potentially large amounts of data being passed from object to object, a reference can be made via the Container Shortcuts.

Return to the project tree by closing the Container pane.

4.8.2 CREATING THE REPORT

```
🐾 First Action    Print Graphic
   ⊞ 🖨 Printer = mailroom printer    Printer
   □ 📄 Pages    Form
      □ 🔲 Layout    Layout
         □ 🖼 Graphic    Binder
            ⊞ A  First Page    Text Box
```
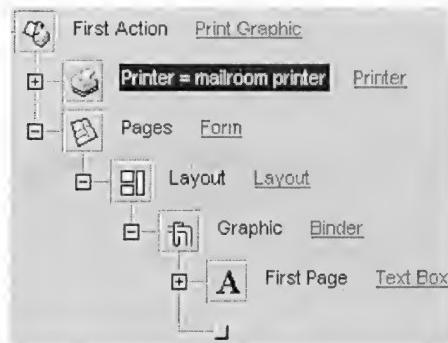
Expand the Print Graphic object.

Change the Reporter object (labeled 'Pages') to a Form object (from the Forms category).

Click on the Form object to open the Pages form editor. Use the Object tool to define an area on the form that is the same size as the page.

When the Create Object dialog box is displayed, select the Layout object from the Forms category. Return to the project tree. Expand the Form and the Layout objects.
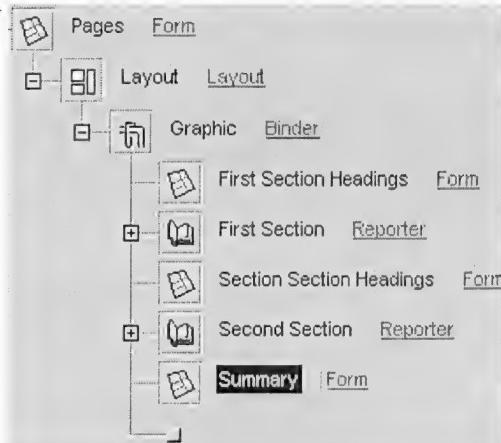
Delete the Text Box object below the Binder object.

```
📄 Pages    Form
   □ 🔲 Layout    Layout
      □ 🖼 Graphic    Binder
            📄 First Section Headings    Form
         ⊞ 📖 First Section    Reporter
            📄 Section Section Headings    Form
         ⊞ 📖 Second Section    Reporter
            📄 Summary    Form
```

Click on the add-point below the Binder object and add a Form object and a Reporter object (both found in the Forms category).

Add a second Form and Reporter object, and then add another Form. Label these according to the section of the report they will create (as shown in the snapshot to the left).

We will now look at creating each section of the report in turn.

4.8.2.1 FIRST SECTION OF REPORT

Speeding Report

Name   Date   Penalty
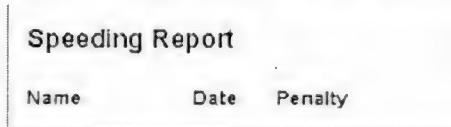
Click on the First Section Heading Form object and define the headings as shown to the left on the First Section Headings form, using the Text tool.

Select Options from the View menu and select the Paper Size tab. Click the Shrink to Fit button and uncheck the Auto-size check box. Close the First Section Headings form editor pane to return to the project tree.

Next, expand the Reporter object (labeled 'First Section').

First Section   Reporter

Data   Walker

Container = Normal.RECORDSET   Container

The first child object (labeled 'Data') specifies what data to loop round. Change this Data object from a PageBank object to a Walker object (look under the Containers category).

Expand the Walker object.

The Bag child object of the Walker specifies at what level in the bag's data structure it is to loop through. Right-click on the child Bag object and go to the Containers tab, where a Container shortcut can be specified. We need to loop round all the children under the RECORDSET folder, so select RECORDSET from the tree.

Click on the First Section Reporter object and click on the Map button. The screen splits into two panes-the Data for Graphic data palette and the Graphic form.

Map across the Surname, Forename, Date and Penalty description to the top of the form, by dragging and dropping the relevant fields from the Data for Graphic pane (using the left mouse button).

Value
  Date 21/06/98
  Method camera
  Road High Street
  Town Aldershot
  Forename Alex
  Surname Wright
  Speed 43
  Speed Limit 30

Wright   Alex   21/06.98   3 penalty points £40 fine
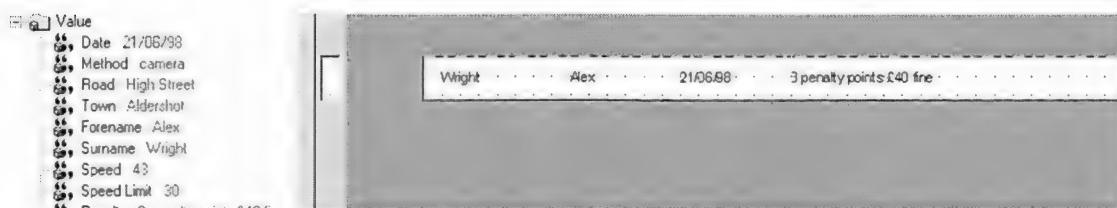
Note: You can check the appearance of the report by opening the original form (i.e. the Form object just below the Printer object).

Next, select Options from the View menu and select the Paper Size tab. Click the Shrink to Fit button and uncheck the Auto-size checkbox.

Close both panes to return to the project tree.

4.8.2.2 SECOND SECTION OF REPORT

Method	Town		Speed	Speed Limit

Open the Second Section Heading Form editor and define the headings shown alongside.

Select Options from the View menu and select the Paper Size tab. Click the Shrink to Fit button and uncheck the Auto-size check box. Close the form and return to the project tree.

Next, expand the Reporter object labeled 'Second Section'. As in the First Section, change the Data object from the PageBank to a Walker and specify the RECORDSET as the Container Shortcut.

Click on the Reporter and press the Map button. Map across the Method, Town, Speed and Speed Limit to the top of the form. Next, select Options from the View menu and select the Paper Size tab. Click the Shrink to Fit button and uncheck the Auto-size check box. Return to the project tree.

4.8.2.3 CREATING THE SUMMARY SECTION

The summary section will contain the sentence "This report contains x speeding offenses, y were caught by cameras." The values x and y need to be defined and will be calculated by building a branch that counts bits of the data. This can be achieved by using a combination of the Memory, Scan and Add objects.

On Job   Memory

Items   Folder

314 | Total Records = 0   Number

314 | Camera Total = 0   Number

Definition   Action List

Scan   Scan

Data   Walker

Container = Normal.RECORDSET   Contai

Action   Action List

First Action   Print Graphic
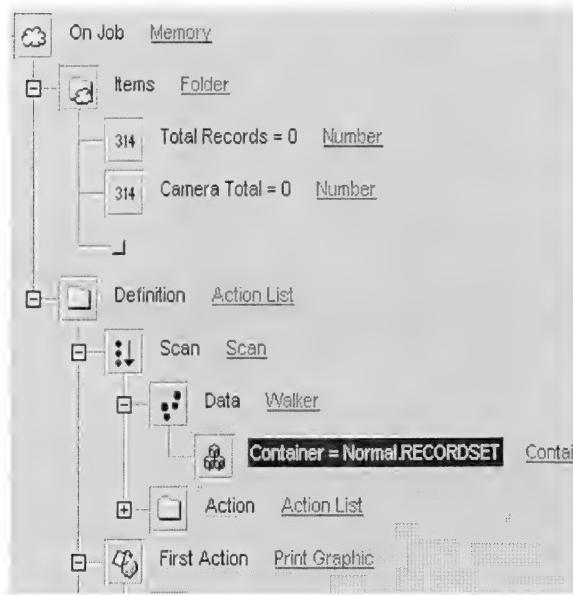
Change the On Job folder to a Memory object (found in the Core category).

Click on the add-point under the Items folder and select a Number object from the Core category. Rename the Number object as "Total Records". Add a second Number object, this time labeling it "Camera Total". These will be used as counter objects.

Expand the folder labeled Definition and add a Scan object (found in the Core category) under the On Job folder and move it above the Print Graphic using the blue up arrow in the toolbar.

Change the child Data object under the Scan object to a Walker (under the Containers category) and it's child object, Bag, should reference the RECORDSET Container Shortcut. This again highlights to loop round all the child folders of RECORDSET, i.e. Loop round each Record.

We have specified what we want to loop round, next we must define our calculations under the Action folder. The first, and simpler of the two calculations is the total number of offenses.

On the add point under the Action List object (labeled 'Action'), add a Copy Number object (look under the Core category). The Copy Number object will be a counter, which will increase by one each time the scan loops around. Expand the Copy Number object to display two Number objects.

Change the first Number object labeled 'From', to an Add object (found in the Core category). Expand the Add object.

⬜	Action	Action List	
⊞ ✍	First Action	Print Graphic	
⊟ 21	Copy Number	Copy Number	
⊟ ➕	From	Add	
🔗	First = On Job.Total Records	Container Sh〔	
314	Second = 1	Number	
🔗	**To = On Job.Total Records**	Container Shortcut	

We are going to add one to the counter, Total Records, which we defined under the Memory object earlier. Therefore, change the First number (under the Add object) to reference this "Total Records" object by right clicking and selecting Total Records from the Containers tab (under On Job).

Click the Number object labeled 'Second' and enter 1.

Change the Number object labeled 'To' to the Total Records Container Shortcut.

We are now incrementing the value 'Total Records' and putting the result back into 'Total Records'. This will occur five times in this example. As this Scan is placed higher on the branch than the Print Graphic, the Total Records counter will reach 5, when it will be used by the Summary section of the Print Graphic.

The second Action we need to define involves calculating the number of offenses picked up by the speed camera. In much the same way as calculating the Total Records, we use a counter system, however, this time it is only incremented if certain conditions are met, i.e. "Is the Method element's value 'camera'?".

? Create a second child under the Action object of type Decision (look under Core) and expand the new object. A Decision allows us to run an Action if a specified test returns 'Yes'. The test will be a Text Match of the text "Camera". If true (Yes), we increment the Camera Total counter.

⊟ **?**	Decision	Decision	
⊟ ✍=	Trigger Tree = Yes	Text Match	
🔗	First = Scan.Method	Container Shortcut	
❝❞	Second = camera	Text	
⊟ 21	When Triggered	Copy Number	
⊟ ➕	From	Add	
🔗	First = On Job.Camera Total	Container Sh〔	
314	**Second = 1**	Number	
314	To = 0	Number	
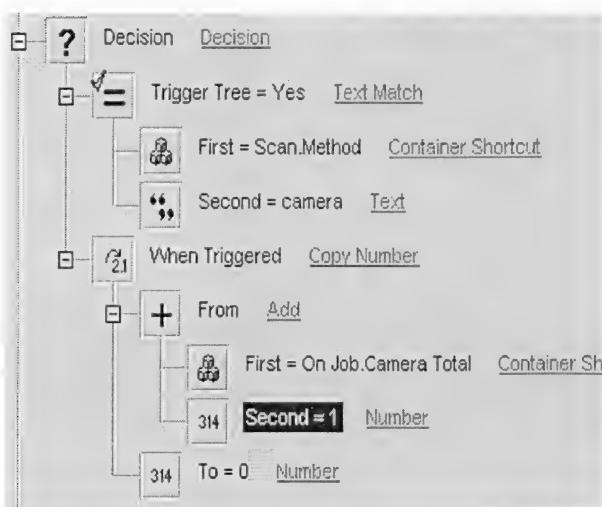
This is achieved by changing the All Of object labeled 'Trigger Tree' to a Text Match object (look under Core).

Its first child (a Text object) should be changed to the Container Shortcut, Method. This is accessible from the Containers tab in the Change Object dialog (select Scan from the Look in object combo box).

The second child of the Text Match object should be the word "camera".

We now need to specify what happens if we find a "camera" record. In a similar vein to the Total Records case, we increment a counter. This time the Camera Total counter.

Change the Action list object labeled 'When Triggered' to a Copy Number object and follow the instructions as before for the Total Records, remembering to reference the Camera Total instead.

Finally, we need to ensure that the value of both counters are reset to zero before the Scan is run. Add another two separate Copy Number objects above the Scan object which copies the value zero into the Total Records and Camera Total container shortcuts. To achieve this, add the objects to the add-point and use the blue arrows tools to move the objects up the tree.
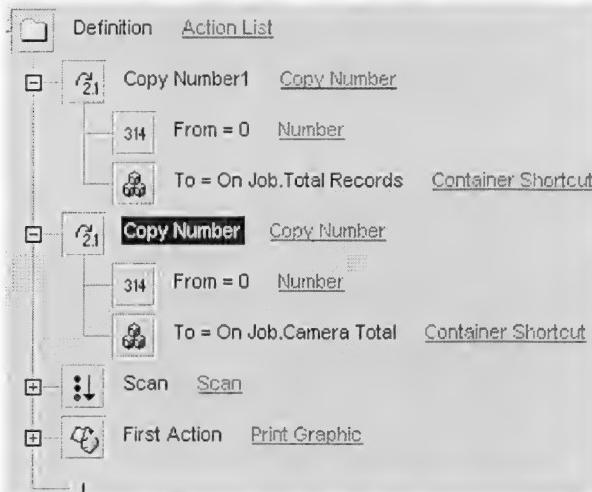
The final section of the report can now be created.

Open the Form object labeled 'Summary', and draw a Text Box. Enter the text "This report contains ", with a space at the end of the text. Right-click on the Text Box and select Explore. Expand the new Text Box object on the tree and change the Text object to a Sentence object (found in the Core category). Expand the Sentence object.

Click on the add-point below the Sentence object and select the Number To Text object (found in the Core category). Expand this Number To Text object and change the Number object below it to the counter, Total Records, container shortcut (in Change Object look under the Containers tab and under the On Job in the Look in combo box).
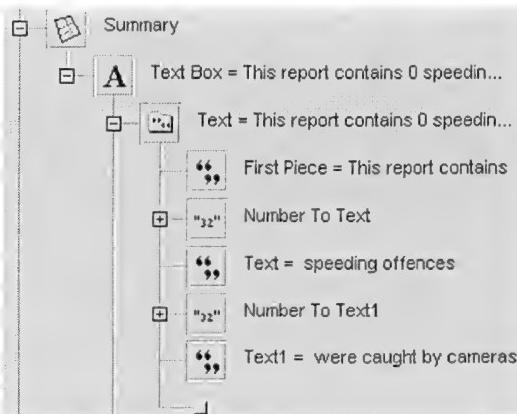
Add a Text object to the Sentence and enter the text " speeding offenses ", with a space at either end of the text.

Add another Number To Text object (in the Core category) under the Sentence, expand it and change the Number object below it to the counter, Camera Total container shortcut (in Change Object dialog, under the Containers tab and under the On Job in the Look in object combo box).

Add a final Text object with the text " were caught by cameras." with a space at the beginning of the text.

Finally, click on the Summary form editor (to enable it) select Options from the View menu and select the Paper Size tab. Click the Shrink to Fit button and uncheck the Auto-size check box.

The report is now complete.

Speeding Report

Name		Date	Penalty
Wright	Alex	21/06/98	3 penalty points £40 fine
Mall	Stuart	30/06/98	3 penalty points £60 fine
Fuller	Karen	03/07/98	6 penalty points and 1 year ban
Cassidy	Steve	08/07/98	5 penalty points and 8 month ban
Weight	Mick	10/07/98	8 penalty points and court action to follow

Method	Town	Speed	Speed Limit
camera	Aldershot	43	30
camera	Alton	90	70
officer	Basingstoke	85	70
camera	Fleet	45	30
officer	Alton	75	60

This report contains 5 speeding offences 3 were caught by cameras

4.9 ADDING HEADERS AND SIGNATURES TO PRE-FORMATTED DOCUMENTS

FormScape can be used to add headers and signatures to documents that do not need any other processing i.e. the format of the document is already suitable for printing. For example, users could produce a letter in Microsoft Word and use FormScape to add the company letterhead or their signature.

4.9.1 CHANGING THE PRINTER DRIVER
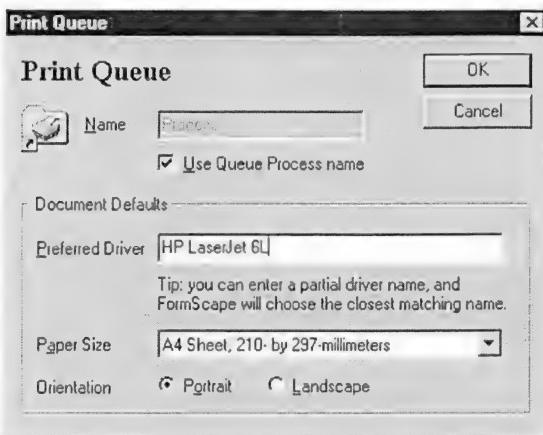
Create a new Queue Process.

The first consideration is the Printer Driver. By default FormScape will use a Generic/Text only driver, but for this Queue Process we need to use the Printer Driver for the final destination printer so that Microsoft Word uses the correct settings for the document.

To change the printer driver, click on the Print Queue object and enter the name of the printer driver into the box labeled 'Preferred Driver'.

The Paper Size and Orientation defined here determine the document defaults of the printer that is created.

Click OK to return to the Project tree.

4.9.2 ADDING THE OVERLAY CHOOSER
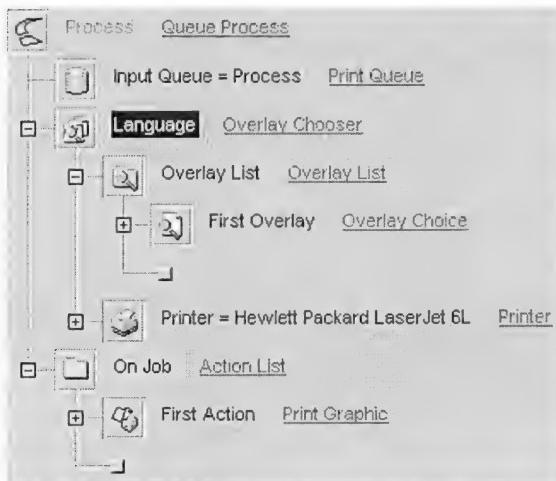
Change the Language object to an Overlay Chooser object (found in the Printing category).

Expand the Overlay Chooser. The Overlay Chooser has a list of Overlays and a Printer object where the output is directed.

4.9.3 CREATING THE LETTERHEAD

First, we will setup an Overlay Choice to print a letterhead.

Rename the Overlay Choice object (labeled 'First Overlay') to 'Letterhead'. Click on the Overlay Choice object to display the Overlay Choice dialog.

Overlay Choice   ×

Trigger   @letterhead@   OK

Macro ID 22131   Cancel

☑ Remove trigger

☐ Relative positioning
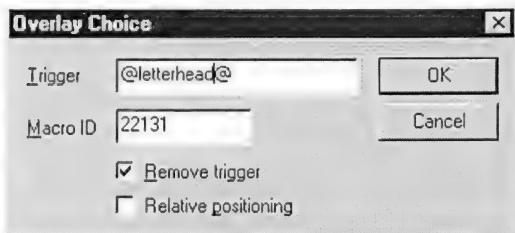
Change the trigger to "@letterhead@" and click on the OK button. The Overlay Chooser will check the data for this text and if it is found it will be removed from the data and the letterhead defined in the form below will be printed. Note that the trigger must use a printer resident font (such as Courier) to be recognized.

Expand the Overlay Choice object and click on the Form object to design the letterhead.

4.9.4 CREATING THE SIGNATURE

Next we will add an Overlay Choice object to place a signature on a document.

Overlay Choice   ×

Trigger   @signature@   OK

Macro ID 22132   Cancel

☑ Remove trigger

☑ Relative positioning
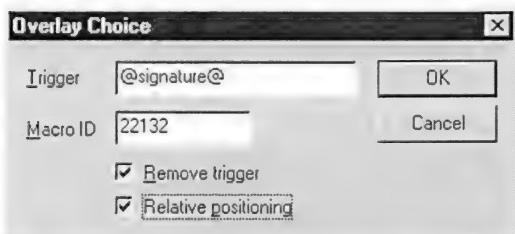
Click on the add point below the Overlay List object and select Overlay Choice (in the Printing category). Rename the Overlay Choice object "Signature".

Click on this Overlay Choice object and change the trigger to "@signature@". Select the Relative positioning checkbox, because the signature does need to appear relative to the trigger. Click OK to return to the project tree
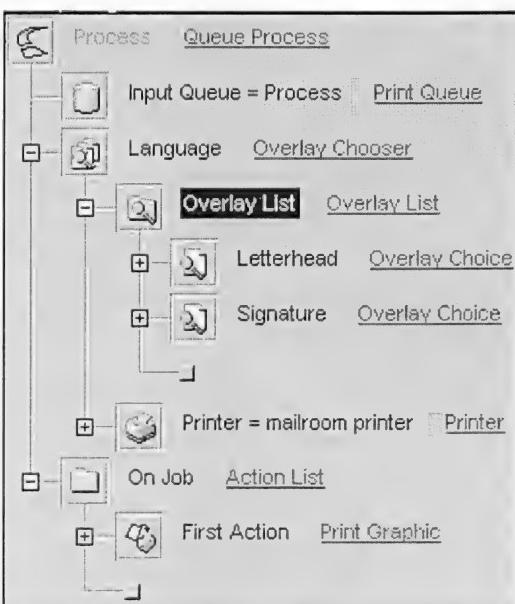
Expand the Overlay Choice object and click on the Form object to define the bitmap of the signature using the Picture tool on the Form editing toolbar, importing a bitmap with a scanned image of the required signature.

Process   Queue Process

⬚ Input Queue = Process   Print Queue

⬚ Language   Overlay Chooser

⬚ **Overlay List**   Overlay List

⬚ Letterhead   Overlay Choice

⬚ Signature   Overlay Choice

⬚ Printer = mailroom printer   Printer

⬚ On Job   Action List

⬚ First Action   Print Graphic

Now all print jobs that arrive will be checked for the defined triggers. Regardless of whether the triggers appear or not, the print job will be sent to the printer (the mailroom printer, in this example).

Note: The Print Graphic object that is under the On Job folder can be deleted because it will not be used.

4.10 INSERTING THE CORRECT CONTACT DETAILS ONTO AN INVOICE

The contact details (name and telephone number) that need to be displayed on an invoice are different depending on the customer number. Each customer number starts with either an 'I', 'F' or 'R'. We shall look at three different methods of achieving this by using either the Chooser, Lookup or Read File objects. These examples use the sample data file *Invoice.txt*.

For each of these examples we need to open the Invoice Queue Process (invoices.fsp) and expand the 'Invoice' Print Graphic object. Click on the Reporter object and click on the Map button. The screen splits into a Graphic form editor and Current Page layout.

A On the Graphic form editor, using the Text tool, define a text box for the contact details on the form and press the Enter key to leave the text empty. Right-click on the text box and select Explore to locate the Text Box on the project tree.

4.10.1 THE CHOOSER OBJECT

A Expand the Text Box object and change the Text object below it to a Chooser object (found in the Core category).
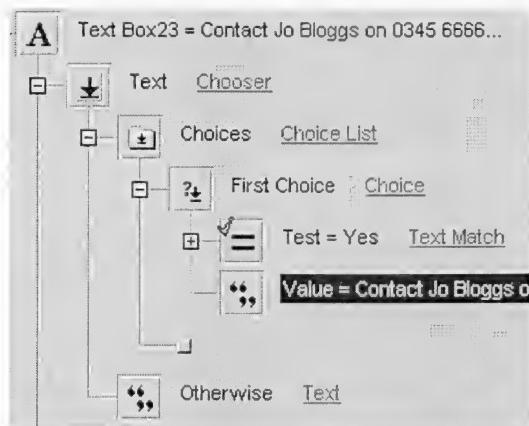
Expand the Chooser object and expand the Choice object below it (labeled 'First Choice').

Ensure that you are displaying the first page of data from the sample file (click on the Reporter object) and draw a box around the letter 'F' at the beginning of the customer number (on the Current Page pane).

Right-click and drag and drop onto the Yes/No object – the icon changes.

Click on the Text object icon labeled "Value" and enter the text "Contact Jo Bloggs on 0345 666666 with any queries".

Click on the add-point below the Choice List object and add another Choice object (in the Core category).

Expand the new Choice object and move to page 2 of the data (click on the Current Page pane to select it and click on the right arrow button).

Right-click on the box you drew earlier (it now contains the letter 'I') and drag and drop it onto the new Yes/No object.

Click the Text object labeled 'Value' and enter the text "Contact John Smith on 0345 777777 with any queries".

Now click on the Text object labeled "Otherwise" and enter the text "Contact Sarah Jones on 0345 888888 with any queries".
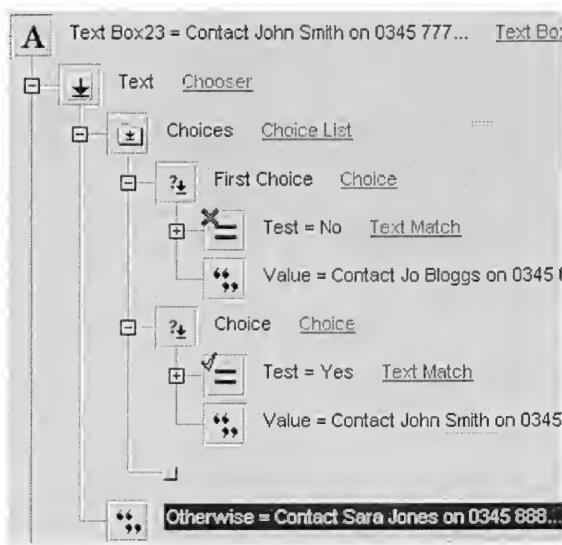
The correct contact details will now be inserted on the invoice, depending on the first character of the customer number.

4.10.2 THE LOOKUP OBJECT

(Before trying this exercise, be sure to re-open the invoices.fsp project, reload the file and recreate the text box).
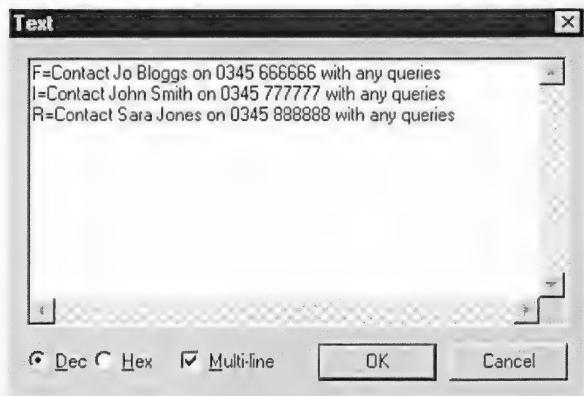
Expand the Text Box, and change its Text object, labeled 'Text' to a Lookup object (found in the Deprecated category).

Expand the Lookup object.

Expand the Tokenizer object below it.

Click on the Text object labeled 'Text', and enter the three lines of text displayed to the left. (To start a new line of text, hold the Control key when pressing Enter).

By default the Text object labeled 'Delimiter' is set to a carriage return and line feed, so these options do not need to be changed.

In the Current Page pane draw a box around the letter 'F' at the beginning of the customer number on page 1 of the data. Right-click and drag and drop onto the Text object labeled 'Key'. The correct contact details will now be inserted onto the invoice.

4.10.3 THE READ FILE OBJECT

For this example we need to create a directory on the C drive called "contacts" and three text files called F.txt, I.txt and R.txt respectively. Each text file should contain the appropriate contact information e.g. 'Contact Jo Bloggs on...' in the F.txt file, 'Contact John Smith on...' in the I.txt file, and so on.
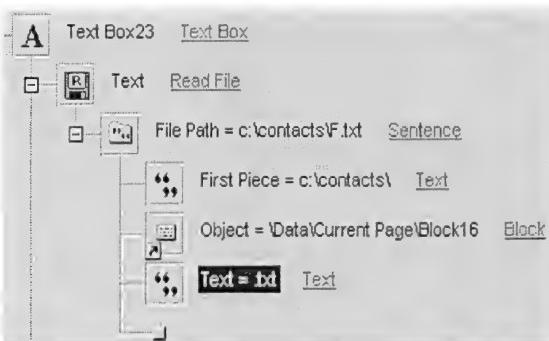
Expand the Text Box object and change its Text object to a Read File object (found in the Files category).

Expand the Read File object and change the Text object below it labeled 'File Path' to a Sentence object (found in the Core category).

Click on the Text object below the Sentence object labeled 'First Piece', and enter the text "C:\contacts\".

Draw a box around the letter 'F' at the beginning of the customer number on page 1 of the sample data, right click and drag and drop it onto the Add point below the Sentence object.

Finally, add another Text object to the Add point and enter the text ".txt". The correct file will now be selected for the invoice.

4.11 USING BRANCHES

4.11.1 USING BRANCHES TO MAKE GLOBAL CHANGES

It is possible to save any object as a branch. If you right-click on any object you will be given the option to Save either 'To the Server' or 'To My Computer'. Once an object is saved to the Server, the branch will become available if you change the type to a compatible object (using the branches tab). In the following example we will create a branch holding the company address and then reference the branch rather than retype the text on every form.

Start a new branch in FormScape Developer. Change the Queue Process object to a Text Box object (found in the Forms category).

Expand the Text Box object, click the Text object and enter the company address (hold down the Control key and press Enter to move to a new line).

Right-click on the Text Box object, select Save As from the menu and select To the Server. Create a new folder called Shared Resources and click on the Save button. In this example, the branch has been named 'Company Address'.

Next we need to use the company address branch within our project.

Open a project where the company address is required. Expand the Print Graphic and Reporter objects, and click on the Form object to open the Graphic form editor.

Select the Text Box tool and define the area on the Form where the address should appear. Press the Enter key to leave the text empty. Right-click on the text box and select Explore to locate the Text Box on the project tree.

Right-click on the Text Box object, and select Change Type.

Click on the Branch tab and go to the Shared Resources folder.

Double-click on the Company Address branch and the file is added to the project tree.
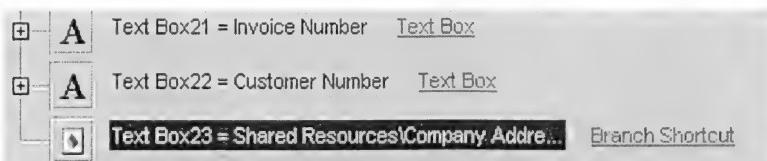
You can use the same procedure to reuse the company address branch in any project. If the company address changes, simply open the Company Address branch and make the changes. When the branch is saved the changes will be effective wherever it is used.

4.11.2 USING INPUTS

It may be desirable to use different point sizes for the company address at different locations. To achieve this you need to setup a Parameter.
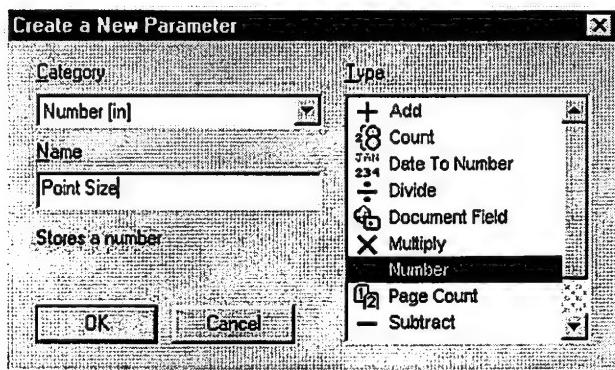

Open the Company Address branch created above.

Right-click on the Number object labeled 'Point Size' and select Change Type from the menu.

Click the Parameter tab. Click the New button. Ensure that the Type is 'Number' and change the Name to 'Point Size'. Click OK twice to return to the tree.

The Number object will now appear as a Parameter object on the tree.
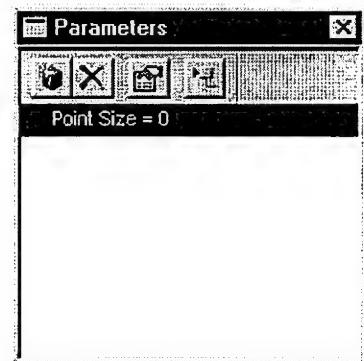
If you want to set a default value for the Point Size, click the Parameters tool on the toolbar to open the Parameters editor.

Select Point Size and click the Edit Default Value button.

Enter the required value (0 in this example) and click on the OK button.

Click on the ⊠ in the upper right corner to close the Parameters window.
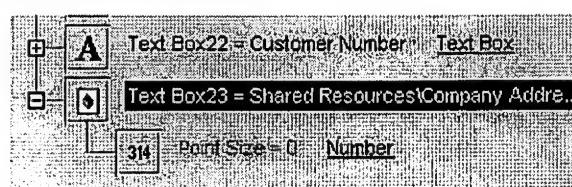
Select Save from the File menu to save the changes.

Now open the Queue Process branch where the Company Address branch was been used. Locate the branch in the tree.

Click on the branch and then click OK (this is to refresh the branch). The branch will now have a Number child object. Enter the required point size into the Number object.

A second parameter could be defined if, for example, the color of the text was to change according to the location. Follow the same procedure as above to create the color parameter, not forgetting to refresh the branch in the Queue Process that uses the Company Address branch.
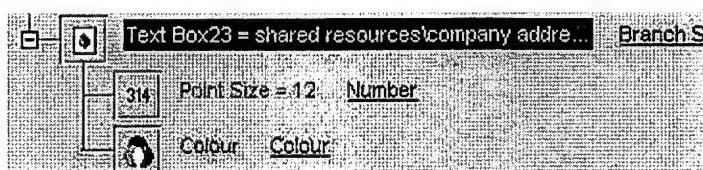
This will give the resulting child objects as shown above.

4.12 CREATING A GENERIC QUEUE PROCESS

As pointed out above, the Processes folder is used to hold branch processes that we want to start automatically. However, there are two reasons why we may not want to save the whole Queue Process in this folder.

The first reason is that when you save a branch process into the Processes folder the process is stopped and restarted. If you are using Print Queues, the Printer will disappear from the printer folder momentarily.

The second reason is that the name of the root folder is computer-dependent, so if you are copying projects between computers either by restoring from backup or using the Administrator module, this Processes folder will not be the active Processes folder.

To deal with both of these issues we will create a generic Queue Process.
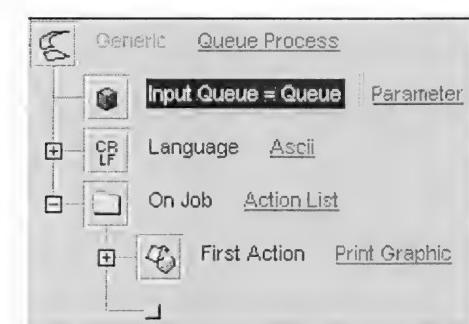
First create a new Queue Process.

Right-click on the Print Queue object labeled 'Input Queue' and select Change Type. Click the Parameter tab and click the New button.

Select Print Queue from the Type list and click OK twice to return to the project tree.

Save the branch to the Server in the Shared Resources folder that was created earlier. The Queue Process in this example has been called 'Generic'.

Next select New from the File menu to create a new Queue Process.

Right-click on the Queue Process object and select Change Type.

Click the Branch tab and double-click on the Generic branch. The branch will display on the tree with the Print Queue object as a child object.

Click on the Print Queue object and enter the name 'New Queue' in the Name box (you may have to uncheck the Use Queue Process name checkbox). Click OK.

Now save this to the Processes folder on the Server with a name 'New Queue'. The process will now be started, and a printer called 'New Queue' will appear in the Printer folder.

4.13 CONVERTING FORMSCAPE OUTPUT TO PDF FORMAT

Third-party software can be used to convert output from FormScape to Adobe's popular PDF format. The steps below explain how to achieve this using Adobe's own PDF Distiller product.

You must install a PostScript driver onto the computer with the FormScape Server. If you are using FormScape Developer remotely from the server you must install the drivers on the remote machines too. Apple PostScript drivers have been found to provide the best PostScript output to use with Distiller.

1. Run the Acrobat Distiller program and select Watched Folders from the Distiller menu. Click on Add Folder to add a folder to the list. Select an existing folder and Distiller creates two sub-folders called *in* and *out*.

2. Start FormScape and create a new project.

3. Change the Print Graphic object to a Print to Stream object. Ensure that the selected printer under the Print to Stream object is a PostScript printer.

4. Change the child Text object to a Write File object (in the Files category).

5. Open a file and map the information onto a form, for example as we did in Tudor1 that was created earlier in this guide (see page 17).

6. Click on the Write to File object – A Save As dialog is displayed. In the *in* folder that was created in step 1, enter a file name with a *.ps* extension. The tree should now look similar to the following:
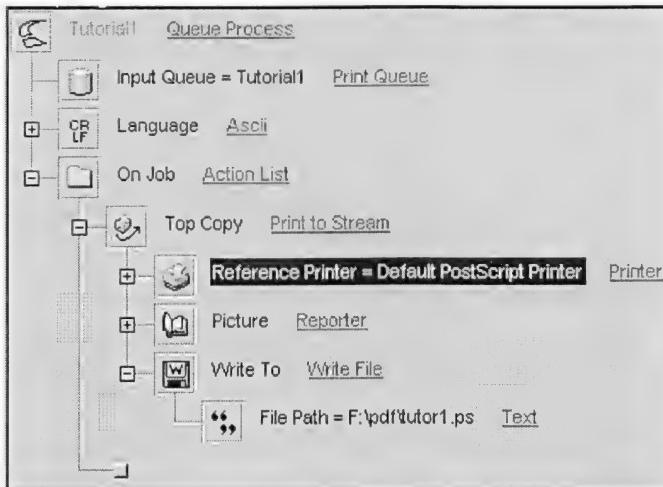
7. Right-click on the Action List object (labeled 'On Job) and select Run from the menu. The file is processed and an Event Log is displayed.

    The Distiller program detects that a file has been created in the watched folder, converts the PostScript file and places the new PDF file in the *out* sub-folder. Depending on the Watched Folders Options, Distiller will delete the Postscript file in the Distiller's *in* sub-folder or copy it to the *out* sub-folder.

In FormScape v2 you can send PDF format documents as attachments to e-mail. Use the Check File object to ensure that the PDF file has been created. You can then use the Send Mail object to send the PDF file as an attachment.

APPENDIX A   TOOLBARS AND MENUS

The following pages list and describe the various toolbars and drop-down menus associated with the FormScape Developer screen. Some of these items will already be familiar to you as you have made use of them in the tutorial projects you have created, but the listings below can be used for reference purposes and to supplement your awareness of FormScape's capabilities.

A.1   MAIN TREE TOOLBAR AND MENUS

The Main Toolbar in the Developer module comprises the following buttons and drop-down menus:
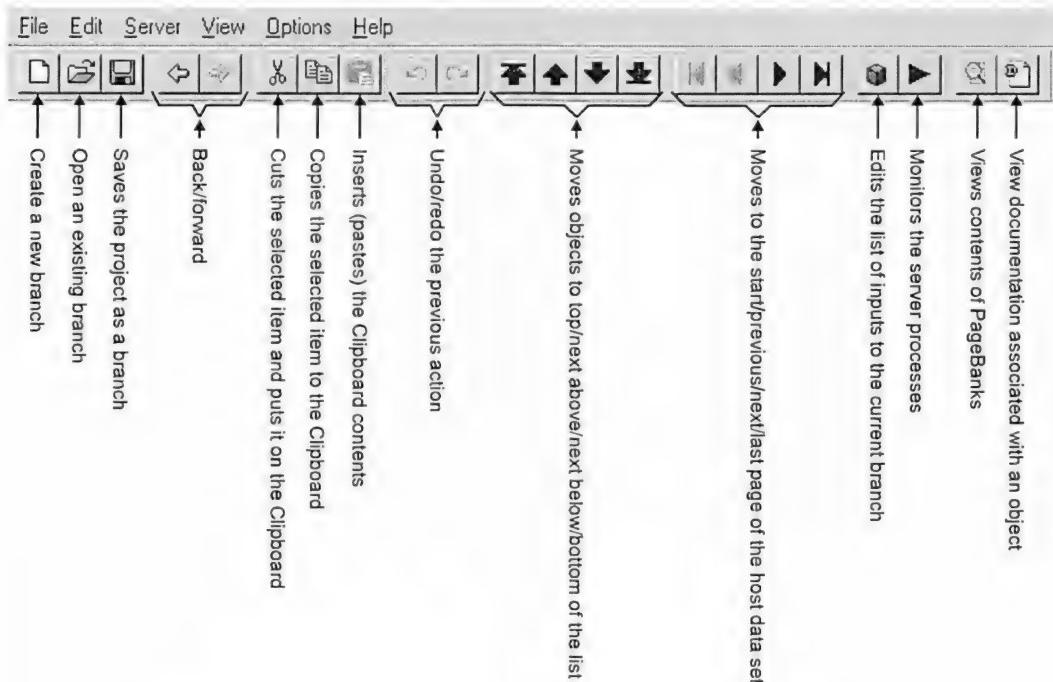
The functions of the buttons (from left to right), with their associated menus and keyboard commands, on this Toolbar are as follows: -

| Button	Function	Menu/keyboard command
New branch	Opens a new branch in Developer: if you are already editing a branch you will be prompted to save it. The new branch has a default Queue Process called Process : it is recommended to save the new branch immediately with a meaningful name	File|New Alt F N Ctrl N
Open branch	Opens the project browser to find an existing FormScape branch file. Browse through the branches until you find the one you wish to edit : when you select it, it is copied from the server to the Developer	File|Open Alt F O
Save branch	Opens the project browser: enter a name for your branch. The current branch is then saved to the server using that filename	File|Save Alt F S Ctrl S
Back/forward	Arrow tools allow you to flick back/forward through the branch files you have been using	File|Back Alt F B File|Forward Alt F F
Cut/copy/ paste	Cuts the selected (highlighted) item and places it on the Windows clipboard/copies the selected item to the Clipboard/pastes the Clipboard contents to the selected location on the Developer tree. These buttons should be used with caution to consider the possible effects of any existing links between the item and other objects in the project	Edit|Cut Alt E T Ctrl X Edit|Copy Alt E C Ctrl C Edit|Paste Alt E P Ctrl V
Undo/redo	Remembers the last actions you performed and enables you to undo/redo them	Edit|Undo Alt E U Ctrl Z
Move Object	Allows you to move the selected (highlighted) object to the top, next above, next below or bottom of the level on the tree	
Change Page	Allows you to 'flick' through a set of pages in a PageBank by moving to the start, previous, next and last page in the data set. Note that these four buttons are only operative if the selected item is a data set object e.g. a page mapper	
Inputs	Opens the Inputs dialog box enabling you to edit/remove inputs for the current branch. Using inputs allows you to share common data between different branches	View|Input Alt V I
Processes	Opens the Monitor dialog box enabling you to control and monitor the processes in your project on the server	Server| Processes Alt S P
PageBank Viewer	Opens the PageBank Viewer screen enabling you to check the contents of a PageBank used in your project, and to delete, load and save PageBanks manually	View| PageBankViewer Alt V P
View documentation	Views any documentation comments associated with a selected object on the tree. Any comments added will be displayed in green next to the object label.	View|Document- ation Viewer Alt V D

The drop-down menus on this Toolbar are as follows: -

File menu

File

New	Ctrl+N
Open	▶
Save As	▶
Save	Ctrl+S
Print Tree	
Back	
Forward	
Exit	
1 C:\online\percentage function	
2 C:\CUSTS\Me\Mickq&c	
3 invoice print graphic	
4 C:\CUSTS\Me\Mickq&c2	
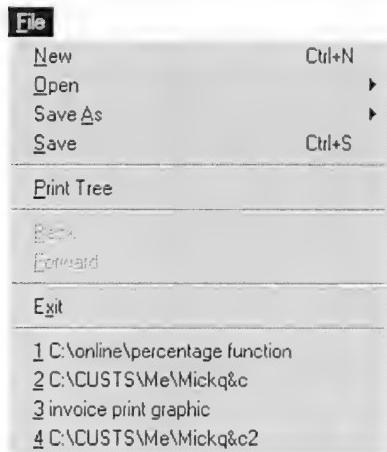
New – Opens a new branch file
Open – Opens a dialog box to select the source of an existing file from either the FormScape server or your computer's hard disk
Save As – Opens a dialog box to save the file to either the FormScape server or your computer's hard disk.
Save – Updates your current file
Print Tree – Prints a graphic showing the tree structure as currently displayed (any displayed documentation comments will also be printed)
Back – Selects the previous branch file
Forward – Selects the next branch file
Exit – Quits Developer (you will be asked to save your project before shutting down)

Edit menu

Edit

|	
Undo	Ctrl+Z
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Find...	Ctrl+F
Find Next	F3
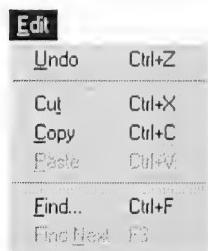
Undo – Undoes previous action
Cut – Removes selected item and places it on the Windows Clipboard
Copy – Copies selected item to the Windows Clipboard
Paste – Inserts the contents of the Clipboard to the selected location

Find/Find Next – Opens a dialog box allowing you to select an object by name and then find it or its next occurrence on the project tree

Server menu

Server

Processes...
Reprint...
Disk Space...
Viewer Users...
Log off
Change Server...
Server Settings...
Stop communication

Processes – Shows a dialog box with the activity of the server. Used to start/stop individual processes in your application
Reprint – Opens a dialog box to gain access to the Reprint Manager features
Disk Space – Opens a dialog box to display disk space usage and to allow you if desired to set a warning message when the disk space usage reaches 80% of its allocated limit (the default limit is 100Mb). If the total space used exceeds the limit, stored documents will start to be deleted
Viewer Users – Displays a list of current Viewer users
Log off – Disconnects from currently-selected server
Change Server – Log on to a different server
Server Settings – Displays the current server settings. Used to control the FormScape working path and its logging functions
Stop communication – Use this to stop Developer searching for the server. A small red cross icon at the bottom right lights up when Developer is communicating with the server. Click on this icon to stop communication

View menu
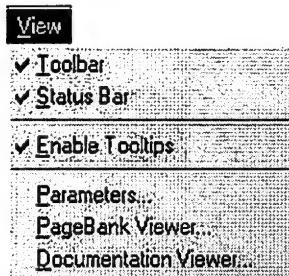
Toolbar – Toggles toolbar at top of screen on/off

Status bar – Toggles status bar (the information line at the bottom of the screen) on/off

**Enable Tooltips** – Displays/hides Tooltips labelling the functions of the buttons

Inputs – Opens an Inputs dialog box to add, edit or remove inputs for the current branch

PageBank Viewer – Displays the current PageBank contents.

Clicking on this option (or on the View PageBank button on the main tree toolbar at the top of the screen) produces a window with a list of all the PageBanks in the current branch with data in them. If you click on one of these, the data on the current page is displayed and you can use the arrow keys to move forward/backward to view the pages in the selected PageBank. Zoom in/out buttons allow you to view the page at three different magnifications. You can delete a PageBank that is no longer required by selecting it in the list and pressing the keyboard DEL key to remove it.

Documentation Viewer – Opens a window to allow you to insert a documentation comment alongside an object's label

Options menu
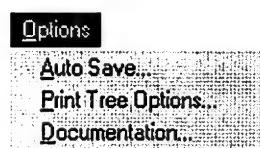
AutoSave – Enables/disables the AutoSave function, which automatically saves a copy of your current branch at regular preset intervals

Print Tree Options – Sets the zoom level for printing the tree view

Documentation – Hides/displays a documentation comment added to a selected object on the tree. A further option allows you to display only a specified number of characters of the comment (in a range of 5-200 characters)

Help menu
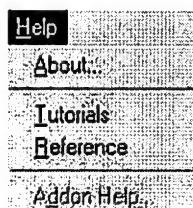
About – Displays version details

Tutorials – Opens Windows Helpfiles

Reference – Displays complete list of FormScape icons arranged alphabetically

Addon Help – For access Helpfiles for 'add-on' modules.

A.2   GRAPHICS (FORM EDITING) TOOLBAR

The Graphics Toolbar in the Form editing screen (revealed by clicking on the Form icon) comprises the following buttons: -


The following labels point to the toolbar buttons:

'Pick tool

Zoom in/out

Undo/redo the previous action

Insert text on the form

Draw a line on the form

Draw a rectangle (box) on the form

Draw an ellipse on the form

Draw an arc on the form

Insert a picture (*.bmp graphic format) on the form

Insert a barcode on the form

Insert a PDF barcode on the form

Insert any other item on the form from a list

Toggles the gridlines on the form on/off

Toggles the snap-to-grid display on the form on/off

Toggles the vertical/horizontal alignment guidelines on the form on/off

Toggles the rulers and dimensions of a selected object on the form on/off

Groups selected items on the form together as a single item

Ungroups items from a selected group on the form

Carries out a test print of the form

The functions of the buttons (from left to right) on this Toolbar are as follows: -

Button	Function	Menu/ keyboard command
'Pick' tool	Click once on the object you wish to select. If you pick the wrong object because it is beneath a number of overlapping objects press Shift and click again to cycle through them until you find the desired object	Tool|Pick Alt T P
Zoom in/out	Enlarges/reduces the size of the mapped page. You can also zoom by pressing the Alt key + double click on the left mouse button (zooms in), or Alt + double click on the right mouse button (zooms out)	Tool|ZoomIn/ Out Alt T I Alt T Z
Undo/redo	Remembers the last actions you performed and enables you to undo/redo them	Edit|Undo/Redo Alt E U Ctrl Z Alt E R

|		
Text	Inserts a text box on the form. Text can be rotated, justified, resized and the font and color changed	Tool|Text Alt T T
Line	Inserts a line on the form. You can select a line style, color and thickness, and the line can be constrained to vertical or horizontal by pressing the Shift key whilst drawing it	Tool|Line Alt T L
Box	Inserts a rectangle on the form. You can choose a border style, color and thickness, and a color fill for the box. The box can have a shadow effect added, rounded corners, and be made transparent (i.e. a selected color fill disabled), and can be constrained to a square by pressing the Shift key whilst drawing it	Tool|Box Alt T B
Ellipse	Inserts an ellipse on the form. You can choose a border style, color and thickness, and a color fill for the ellipse. The ellipse can have a shadow effect added, and made transparent (i.e. a selected color fill disabled), and can be constrained to a circle by pressing the Shift key whilst drawing it	Tool| Ellipse Alt T E
Arc	Inserts an arc on the form. You can select a line style, color and thickness for the arc. The arc can be constrained to a quadrant of a circle by pressing the Shift key whilst drawing it	Tool|Arc Alt T A
Picture	Inserts a color or black/white graphic file (must be in Windows .BMP format) on the form. You can resize the image within FormScape and choose whether to retain the original aspect ratio or stretch the image horizontally or vertically	Tool|Picture Alt T C
Barcode	Inserts a barcode on the form. You can set the value of the text by editing the Barcode object on the tree. To map data onto the page as a barcode, drag it as a normal mapping step. Barcodes can be printed using Code 39, Code 128B, Interleaved 2 of 5, Codabar, UPC A, EAN 13, EAN 8 or Postnet symbologies, and can be printed (in black only) horizontally, vertically or mirrored, with/without checksums and with/without human readable text	Tool|Barcode Alt T R
PDF barcode	Inserts a PDF (Portable Data Format) barcode on the form. This is a special type of barcode from Symbol Technologies and enables you to encode up to approx. 1kb of data	Tool|PDF Barcode Alt T F
Other object	Inserts any printable object on the form selected from a list in an object selector box. The list comprises pictures, barcodes (including PDF format), dynamic pictures, Reporter, Page Setup, Text Box, Gradient, Form, OLE Object, Memory, If, Binder, Layout, Chooser, Set Item, Stored Document and Variable objects. For more details about specific objects in this list consult Appendix B	Tool|Other Alt T O
Major Grid	Shows/hides the major gridlines (vertical/horizontal dotted lines) on the form	View|Show Grid|Major Grid Alt V G M
Minor Grid	Shows/hides the minor grid (the snap grid points) on the form	View|Show Grid|Minor Grid Alt V G I
Alignment guidelines	Shows/hides the alignment guidelines on the form. You can set these to any desired position on the form by clicking on the horizontal/vertical rulers and dragging	View|Show Align Guides Alt V A
Rulers/ dimensions	Shows/hides the horizontal/vertical rulers on the form and the dimensions of a selected object on the form	View|Show Rulers Alt V R
Group/ ungroup	Groups/ungroups selected objects on the form as a single item/separate items	Edit|Group/ Ungroup Alt E G Alt E N
Test print	Prints the currently displayed form to a printer visible to your computer	Tool| Print Page Alt T N

The Toolbar at the top of the screen displays the following buttons depending on whether you now select text editing or box editing on the blank form :-

Text editing

File Edit Format Tool View Help

← Text justified

← Text right aligned

← Text centred

← Text left aligned

← Text unjustified

← Make text coloured

← Make text underlined

← Make text italic

← Make text bold

← Font size (in points)

← Font

← Additional Tools button

Box editing

File Edit Format Tool View Help

← Rotates an arc on the form

← Toggles the box's coloured fill on/off

← Adds a drop shadow effect to the selected box

← Draws rounded corners on the selected box

← Line thickness (in points)

← Line style

← Fills box with colour

← Make box frame coloured

← Additional Tools button

The functions of the buttons on these two toolbars are as follows :-

Button	Function	Menu/keyboard command
Additional tools (More)	Opens a drop-down menu (described below)	
Font	Selects a font name	
Font size	Selects a font size (in points)	
Bold	Makes a selected text bold	Format|Bold Alt O B
Italic	Makes a selected text italic	Format| Italic Alt O I
Underline	Underlines a selected text	Format| Underline Alt O D
Color	Opens a color palette to select a color for the selected text	Format|Fore-ground Color Alt O C
Unjustified text	Removes justification from a selected text	Format|Horizontally|None Alt O H N
Left aligned text	Aligns the text to the left edge of the text box	Format|Horizontally|Left Alt O H L
Centered text	Centers text to the center of the text box	Format|Horizontally|Center Alt O H E
Right aligned text	Aligns the text to the right edge of the text box	Format|Horizontally|Right Alt O H R
Justified text	Justifies selected text between the left and right edges of the text box	Format|Horizontally|Full Alt O H U
Color box frame	Opens a color palette to select a color for the rectangle/ellipse frame (i.e. the outline)	Format|Fore-ground Color Alt O C
Color box fill	Opens a color palette to select a color for the rectangle/ellipse fill (i.e. the inside)	Format|Fill Color Alt O F
Line style	Selects a line style e.g. continuous, dashed, dotted	
Line thickness	Selects a line thickness in points (72 points = 1 inch)	
Rounded corners	Opens a dialog box to allow you to set a corner radius for a rectangle with rounded corners. The Rounded Corners dialog box allows you to specify the size of the corner radius as well as which of the four corners you wish to make rounded	Format|Rounded Corners Alt O O
Drop shadow effect	Opens a dialog box to allow you to produce a further rectangle/ellipse with a colored fill as a 'shadow' to your selected rectangle/ellipse. In this dialog box you can specify the amount of offset of your shadow and click on Set Color to select a color for the shadow from a color palette	Format|Shadow Alt O S
Toggle color fill on/off	Turns a previously selected color fill on/off	Format|Solid Alt O D
Rotates arc	Clicking rotates the selected arc 90° clockwise	Format|RotateArc Alt O A

The Additional Tools button
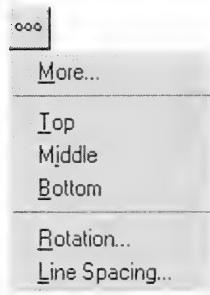
More . . – Toggles between the two toolbars
Top – Moves the text to the top of the text box
**Middle** – Moves the text to the centre of the text box
Bottom – Moves the text to the bottom of the text box
Rotation . . – Opens a further dialog box to allow you to rotate a line or lines of text about its centre point. Note that the rotation is carried out in an **anticlockwise** direction by the specified number of degrees
Line Spacing . . – Opens a further dialog box to allow you to alter the line spacing (for text of more than one line) by specifying a point size e.g. if your text is in a 10 point font and you specify 20 point in the selection box, this will have the effect of 'double spacing' your lines of text

The drop-down menus on either Toolbar are the same, as follows :-

File menu
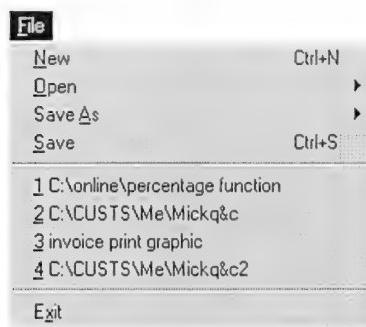
**New** - Opens a branch file
Open - Opens a dialog box to select the source of an existing file from either the FormScape server or your computer's hard disk
Save As - Opens a dialog box to save the file to either the FormScape server or your computer's hard disk
Save - Updates your current file
Exit - Quits Developer (you will be asked to save your project before shutting down)

Edit menu
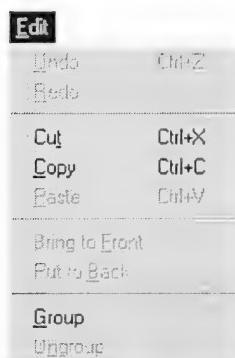
Undo - Undo previous action
Redo - Redo the previous action
Cut - Removes selected item and places it on the Windows Clipboard
Copy – Copies selected item to the Windows Clipboard
Paste – Inserts the contents of the Clipboard to the selected location
Bring to Front – Brings a selected item on the form to the top (front) in a set of overlapping items
Put to Back – Puts a selected item on the form to the bottom (back) of a set of overlapping items
Group – Groups selected items on the form and treats them as a single item
Ungroup – Ungroups the items of a selected group into separate items

Format menu

Foreground Colour - Opens a colour palette to select a colour for the selected text or box frame

Fill Colour - Opens a colour palette to select a colour to fill the selected box

Bold - Makes a selected text bold

Italic - Makes a selected text italic

Underline - Underlines a selected text

Horizontally - Opens a further menu to allow you to align text left, right, center, justified or non-justified

Vertically - Opens a further menu to allow you to justify text to the top, middle (i.e. centered vertically) or bottom of the text box

Rotation - Opens a further dialog box to allow you to rotate a line or lines of text about its center point. The rotation is performed in an anti-clockwise direction from the horizontal by the specified number of degrees

Line Spacing - Opens a further dialog box to allow you to adjust the line spacing of a text by specifying a point size

Rounded Corners - Opens a dialog box to allow you to set a corner radius on a selected rectangle: corners can be independently selected

Shadow - Opens a dialog box to allow you to produce a 'drop shadow' effect on your selected rectangle or ellipse (note that the shadow duplicates precisely the outline of the original rectangle/ellipse - you cannot e.g. have a shadow with rounded corners if your rectangle has square corners). The Set Colour button on this box allows you to choose a color from a palette for your shadow effect

Solid - Turns a previously selected color fill on/off

Rotate Arc – Rotates the selected arc 90° clockwise

Offset Form – Opens a dialog box to allow you to shift (offset) all the items on the form by a vertical and/or horizontal amount specified in two boxes

Tool menu
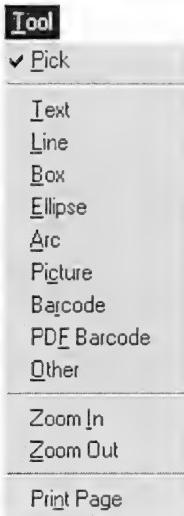
Pick - Allows you to select an existing item on the form

Text - Inserts a text box on the form. Enter the text directly in the window, which then appears, and press Enter at the end of the line of text to quit the window. If instead you press Shift + Enter at the end of the line of text, you can continue onto a second line, and so on

Line - Draws a line on the form

Box - Inserts a rectangular box on the form

Ellipse –Inserts an ellipse based on a rectangle drawn on the form

Arc – Inserts an arc based on opposite corners of a rectangle drawn on the form

Picture - Inserts a picture (Windows .BMP format) on the form

Barcode - Inserts a barcode on the form

PDF Barcode – Inserts a barcode using PDF format

Other - Inserts an item from an object selector box onto the form

Zoom In/Out - Allows you to zoom in/out of the form

Print Page - Performs a test print of the form

View menu
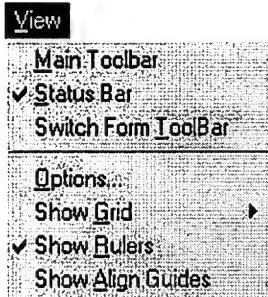
Main Toolbar – Toggles on/off the main toolbar at the top of the Developer screen

Status Bar – Toggles on/off the status information line at the bottom of the screen

Switch Form Toolbar – Toggles between the two toolbars available during form editing

Options – Opens a dialog box with five tabs –

Measurement – Selects measurement units (in/cm), grid size and snap options on the form

Paper size – Allows you to adjust the paper size instead of using the default paper size set by your printer

Background – Allows you to set a background image (e.g. as a watermark) on the form

Zoom – Presets scale factors for zooming in on the form

Font sizing –Permits you to select a font sizing method when importing Version 1 projects (see below)

Show Grid – Toggles on/off the gridlines and snap grid points on the form : a submenu allows you to display/hide the gridlines (Major Grid) and the snap grid points (Minor Grid) on the form

Show Rulers – Toggles on/off the vertical and horizontal rulers on the form, together with the dimensions of a selected item

Show Align Guides – Toggles on/off all vertical and horizontal alignment guidelines on the form

Alignment guidelines are set by clicking on the appropriate horizontal or vertical ruler and then on the form. A coloured vertical or horizontal line appears, which can be moved by clicking on it and dragging to the desired position on the form. This can be repeated for any number of further guides. To remove a particular alignment guideline from the form select it by clicking on it and then either click on Edit|Cut in the drop-down menu or use the Delete key on your keyboard. To align several objects on the form vertically or horizontally, use the CTRL key and mouse button to select the objects so that the last-selected object is the one you wish to align the others to. Click on the **right** mouse button to produce a further menu, and select 'Align' from it. You can then choose whether to align your selected objects to the left, top, right, bottom, centre or middle of the last-selected object.

This further menu, in addition to gaining access to the 'Align' options, also contains a number of options which allow you to cut, copy and paste a selected object, and put it to the back or bring it to the front of a number of overlapping objects. The 'pick up style' and 'apply style' options are used to enable you to make an object have exactly the same attributes as another object on the form. For a text object the attributes which you can 'pick up' by selecting the first object and then 'apply' to the second object by selecting it are the colour, font, font size, line spacing, bold/italic/underline and justification settings, whilst for a box or ellipse object the attributes are the frame colour, thickness, line style, fill colour, and drop shadow and rounded corner settings. For a line or arc object the attributes are the line colour, thickness and line style.

In the Properties option by entering new values for the position of the selected object's edges or its dimensions you can also resize it or reposition it on the form when you click on the Apply button.
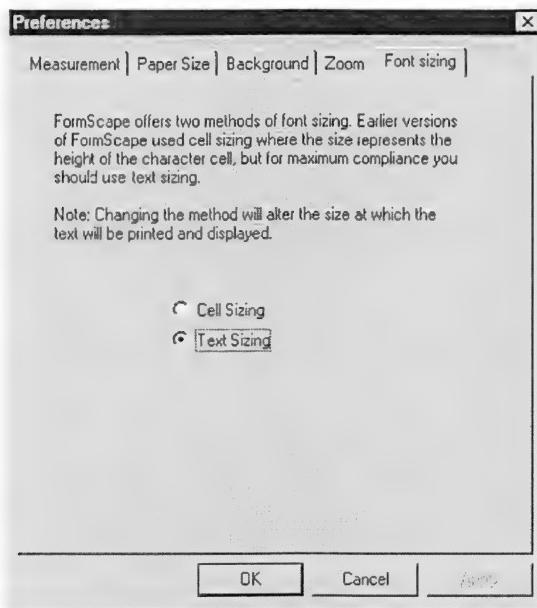
In the Options|Font sizing item you must choose when importing a FormScape project previously developed in Version 1 whether to retain the original character cell size for text (click the 'Cell sizing' button), or change it by clicking the 'Text sizing' button. If you do this, the text will appear and print out more accurately in the Version 2 project.

Help menu

About – Displays version details
Tutorials – Opens Windows Helpfiles
Reference – Displays complete list of FormScape icons arranged alphabetically
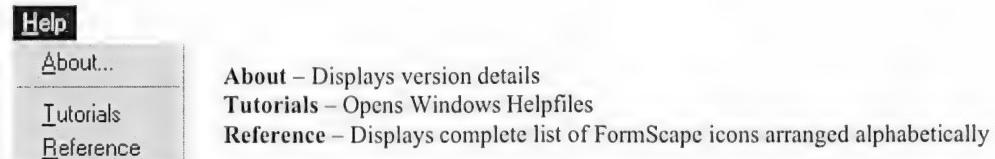
A.3   PAGE PREVIEW (PRINT GRAPHIC) TOOLBAR

The Page Preview Toolbar in the Print Graphic screen (revealed by clicking on the Print Graphic icon) comprises the following buttons: -

The functions of the buttons (from left to right) on this toolbar are as follows: -

Moves to previous/next page

Zoom in/out

Repaginate and redraw the pages in the job
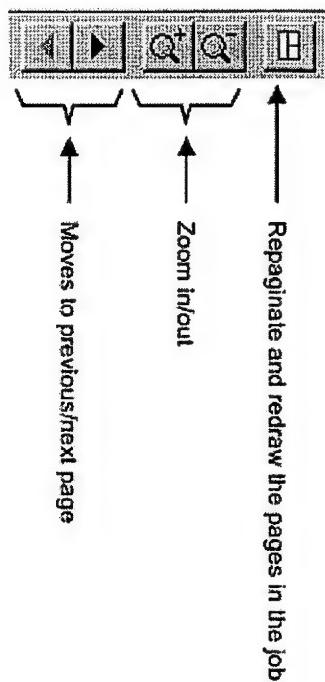
Button	Function	Menu/keyboard command
Previous/next pages	Moves to a full page preview of the previous/next pages of your completed form	Preview| Forward/ Back Alt P F/ Alt P B
Zoom in/out	Allows you to zoom in/out of your page preview	Preview|Zoom In/ Out Alt P I/ Alt P O
Update	Refreshes the page preview when a change has been made to the tree	Preview|Update Alt P U

The drop-down menus associated with this toolbar are as follows: -

File menu   - identical to the File menu previously described

Edit menu
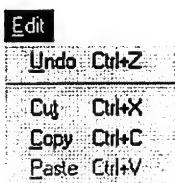
Undo – Undoes previous action
Cut – Removes selected item and places it on the Windows Clipboard
Copy – Copies selected item to the Windows Clipboard
**Paste** – Inserts the contents of the Clipboard to the selected location

Preview menu
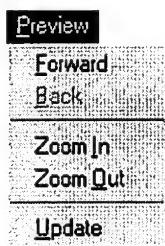
Forward – Moves to a preview of the next page of your print job
Back – Moves to a preview of the previous page of your print job
Zoom In/Out – Allows you to zoom in/out of the form preview
Update - Refreshes the preview when a change has been made to the tree

View menu
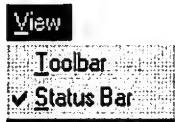
Toolbar – Toggles main toolbar at top of screen on/off
**Status bar** – Toggles status bar (the information line at the bottom of the screen) on/off

Help menu
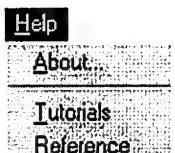
About – Displays version details
Tutorials – Opens Windows Helpfiles
Reference – Displays complete list of FormScape icons arranged alphabetically

A.4 SPLITTER OBJECT TOOLBAR

The Splitter Toolbar (revealed by selecting the Splitter object in the Change Object window and clicking on it) comprises the following buttons :-

The functions of the buttons (from left to right) on this toolbar are as follows :-

Button	Function	Menu/ keyboard command
First/previous/ next/last page	Moves to the first/previous/next/last page in the PageBank	Splitter|First page/ NextPage/ PreviousPage/ Last Page Alt S F/S N Alt S P/S L
PageBank name	Enter or select the PageBank name. Changing this will alter the associated text object on the tree	
Scroll up/down	Moves the top of the page up/down by one line at a time to define where on the data set the page starts (also possible by dragging down the top of the page)	Splitter|Adjust Offset Up/Down Alt S U/ Alt S D
Open Upper/Lower Half	Splits the current pane into two and opens either upper or lower half page mapper	Splitter| OpenUpper Half/ Open Lower Half Alt S O/ Alt S P

The drop-down menus associated with this toolbar are as follows: -

File menu -   identical to the File menu previously described

Edit menu -   identical to the Edit menu previously described

Splitter menu

Splitter
Adjust Offset Up
Adjust Offset Down
First Page
Next Page
Previous Page
Last Page
Open Upper half
Open Lower Half
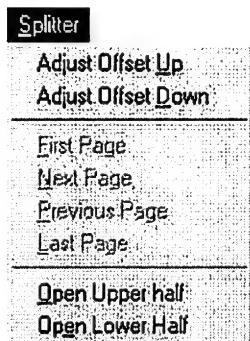
Adjust Offset Up/Down – Scrolls up/down the selected page by one line at a time

First/Next/Previous/Last Page – Move to the appropriate page in the current PageBank

Open Upper/Lower Half – Opens the upper/lower half of the page divided into two by the Splitter object

View menu -   identical to the View menu previously described

Help menu -   identical to the Help menu previously described

A.5   MAPPER OBJECT TOOLBAR

The Mapper Toolbar (revealed by selecting the Mapper object in the Change Object window and clicking on it) comprises the following buttons: -

The functions of the buttons (from left to right) on this toolbar are as follows: -

Button	Function	Menu/ keyboard command
First/previous page	Moves to the first/previous page in the PageBank	Mapper| First Page/ Previous Page Alt M F Alt M P
Page number	Displays or allows entry of a page number in the PageBank	
Next/last page	Moves to the next/last page in the PageBank	Mapper|Next Page/ Last Page Alt M N Alt M L
Zoom in/out	Allows you to zoom in/out of your PageBank view	Mapper|Zoom In/Out AltMI/MO
PageBank name	Displays or allows entry of a PageBank name	
Block name	Displays or allows entry of a text name for the currently selected block in the PageBank	
Cursor position/size of block	Displays either the current cursor position (if no block is selected) or the position and size of the selected block. X is the column number, Y the row number, W the block width in columns and H the block height in rows	

The drop-down menus associated with this toolbar are as follows: -

File menu –   identical to the File menu previously described

Edit menu
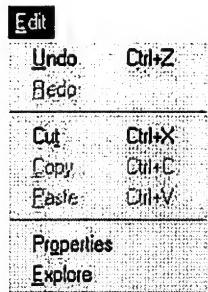
Undo – Undoes previous action
Redo –Redoes the previous action
**Cut** – Removes selected item and places it on the Windows Clipboard
Copy – Copies selected item to the Windows Clipboard
Paste – Inserts the contents of the Clipboard to the selected location
Properties – Allows you to set the space trimming and line merging properties of the selected text box

Explore – Splits the mapper screen in two and opens the tree at the location of the currently–selected block

Mapper menu
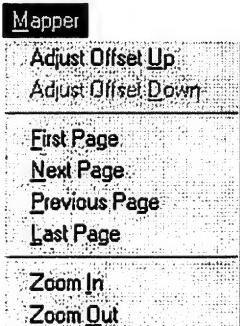
Adjust Offset Up/Down – Moves the top of the page up/down one line at a time
First/Next/Previous/Last Page – Move to the appropriate page in the PageBank

Zoom in/out – Allows you to zoom in/out of the PageBank screen

View menu -   identical to the View menu previously described

Help menu -   identical to the Help menu previously described

APPENDIX B   GLOSSARY OF TERMS

Add point

The empty square box/stub at the end of a branch of the Developer project tree

.BMP

File extension for a Windows bitmap file, the file format required to import graphics into FormScape

COLD

Computer Output to Laser Disc (data storage system)

Conditional text

Text which appears on selected pages of a data sample which have been specified according to a predetermined set of conditions

Conditional processing

The process whereby a text string is made to appear on selected pages of a data set instead of all of them, the pages being selected by a set of conditions previously established in the Developer module

File Queue

Method of collecting data from the network for FormScape systems running in Windows 95 (see also Print Queue)

FormScape Server

This software module processes the print data received from other applications. Once installed and started it requires no further user intervention, and is controlled by the other software modules in the FormScape suite of programs

FormScape Server Lite

This module has the same functionality as the Server module, but is restricted to operating only two printers

FormScape Developer

This is the software module which creates and manages the project for a number of Server modules or sites

FormScape Administrator

This optional software module provides document or FormScape project management from a single centralized point

FormScape Extended
Reprint Manager

This optional software module allows any document processed by the Server module to be retrieved, viewed and reprinted using the FormScape Viewer table

.FSP

File extension for a saved FormScape project : includes everything needed to run the application - forms, data samples, decisions etc.

GUI	Graphical User Interface, the Windows feature which permits you to create complex projects graphically using the Developer module instead of writing lines of programming code
Input page	One of the pages of input data from the host system
Input Queue	The object on the Developer project tree responsible for collecting data from the host system
Language Interpreter	The object on the Developer project tree responsible for taking data from the Input Queue, breaking it down into pages and storing it in a PageBank
Legacy system	An existing data management system on the host system, completely unaffected by the installation of the FormScape software modules
LPD	Line Printer Daemon, the means by which the network system can send (using the TCP/IP protocol) data to a FormScape Input Queue
ODBC	Open Database Connectivity, a system to provide database integration
OLE	Object Linking and Embedding, a method of sharing data between different applications
PageBank	Where the data from the host system is stored in pages in FormScape
PDF	Portable Data Format, a format for creating barcodes from a data set
PCL	Page Control Language, a printer protocol used by FormScape
POP3	Post Office Protocol Version 3, a type of e-mail format
Print Queue	Preferred method of collecting data from the network for FormScape systems running under Windows NT4 (see also File Queue)
Print Graphic	The object on the Developer project tree, which takes the completed form and sends it to a printer
SMTP	Simple Mail Transfer Protocol, a protocol used with e-mail messages
TCP/IP	Transmission Control Protocol/Internet Protocol, the preferred network protocol for FormScape connectivity
WORM	"Write Once, Read Many", a type of storage medium e.g. CD-ROM and some magneto-optical media types

INDEX

F O R M S C A P E®

*e*MPOWERING BUSINESS COMMUNICATIONS

Empower your Business
Communications with FormScape
- within hours

Form®
Scape

Enterprise Output
Management

www.formscape.com

email: info@formscape.com

INTERNATIONAL
OUTPUT NETWORK